



# **Endanflug-Profile in RouGe**

## **Flexible Generierung von Endanflug-Profilen für den Routengenerator RouGe**

Bachelorarbeit  
Hendrik Jüchter

IB-Nummer 112-2014/37  
Zugänglichkeitsstufe A/I



**Institut für Flugführung**  
Direktor: Prof. Dr. Dirk Kügler

## Dokument Information

Zuständiger Projekt- / Dipl.-Ing. Alexander Scharnweber /  
Abteilungsleiter: Dr.-Ing. Michael Schultz  
Zuständiger Autor: Hendrik Jüchter  
Weitere Autor(en):  
Projekt / Zielfeld: Flexible Generierung von Endanflug-Profilen für den Routengenerator RouGe  
Zugänglichkeitsstufe: A / I  
Datei: IB-112-2014-37\_Endanflug-Profile\_001  
Version: 0.01  
Speicherdatum: 2014-10-13  
Gesamtseitenzahl: 42

## Änderungsverfolgung

Version	Datum	Geänderte Seiten / Kapitel	Bemerkungen
0.01	09.10.2014	Gesamtes Dokument	Formatierung der Bachelorarbeit nach Institutsvorlage

© 2014, DLR, Institut für Flugführung, Deutschland

Dieses Werk einschließlich aller seiner Teile ist urheberrechtlich geschützt. Jede Verwendung innerhalb und außerhalb der Grenzen des Urheberrechtsgesetzes ist ohne Zustimmung des DLR, Institut für Flugführung, unzulässig und wird zivil- und strafrechtlich verfolgt. Dies gilt insbesondere für Vervielfältigungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.

# Ehrenwörtliche Erklärung

Hiermit versichere ich, dass ich die vorliegende Arbeit selbstständig angefertigt und keine weiteren als die angegebenen Quellen und Hilfsmittel benutzt habe.

Braunschweig, den

---

Unterschrift

## **Abstract**

The Institute of Flight Guidance of the German Aerospace Center investigates traffic flows in the vicinity of airports using fast time simulation systems. This Terminal Maneuvering Area shows a high complexity and variety of flight routes. Parts of it are the final approaches on the runways of airports. To reduce the high expenditure of the modelling of landing routes, a concept to the adaptable generation of final flight profiles for the route generator RouGe should be developed and be implemented within the scope of this thesis. RouGe generates air traffic scenarios from real flight schedule data.

Trough the development of a modelling module for final flight profiles, an individual landing route for every flight of a scenario does not have to be edited any more by hand. The module enables the user to extend the flight schedules with information about runways and to generate adaptable final flight profiles. The direct approach, trombone approach and point-merge approach can be generated by suitable parameterization. Now less time is required due to the automation of the modelling process for landing procedures within fast time simulation close to reality.

## **Zusammenfassung**

Am Institut für Flugführung des Deutschen Zentrums für Luft- und Raumfahrt werden mithilfe von Schnellzeitsimulationen Verkehrsabläufe im Nahbereich von Flughäfen erforscht. Dieser Nahbereich weist eine hohe Komplexität und Vielfalt an Flugrouten auf. Ein Teil davon sind die Endanflüge auf die Landebahnen der Flughäfen. Um den hohen Aufwand der Modellierung realitätsnaher Anflugrouten zu verringern, soll im Rahmen dieser Bachelorarbeit ein Konzept zur flexiblen Generierung von Endanflug-Profilen für den Routengenerator RouGe entwickelt und implementiert werden. RouGe erzeugt Flugverkehrsszenarien aus realen Flugplandaten.

Durch die Entwicklung eines Modellierungsmoduls für Endanflug-Profilen muss nicht mehr manuell für jeden Flug eines Szenarios eine individuelle Anflugroute eingearbeitet werden. Das Modul ermöglicht es dem Anwender, die Flugpläne um Informationen zu Start- und Landebahnen zu erweitern und flexible Endanflug-Profilen zu erzeugen. Durch geeignete Parametrisierung können der direkte Anflug, der Trombone Anflug und der Point-Merge Anflug betrachtet werden. Durch die Automatisierung des Modellierungsprozesses für Anflugverfahren wird nun weniger Zeitaufwand für die Modellierung einer realitätsnahen Schnellzeitsimulation benötigt.

# Inhaltsverzeichnis

Abkürzungsverzeichnis . . . . .	7
Abbildungsverzeichnis . . . . .	8
Vorwort . . . . .	1
<b>1 Einleitung</b>	<b>2</b>
1.1 Aufgabenstellung . . . . .	2
1.2 Hintergrund . . . . .	2
1.3 Motivation . . . . .	3
<b>2 Grundlagen</b>	<b>4</b>
2.1 Bedeutung der Simulation . . . . .	4
2.2 Technische Grundlagen . . . . .	5
2.2.1 Entwicklung . . . . .	5
2.2.2 Routengenerator (RouGe) . . . . .	5
2.2.3 Auswertung . . . . .	6
2.3 Luftraumstrukturen . . . . .	7
2.4 Endanflugverfahren . . . . .	8
2.4.1 Direct Approach . . . . .	8
2.4.2 Trombone Verfahren . . . . .	9
2.4.3 Point-Merge Verfahren . . . . .	10
<b>3 Konzept Endanflug-Profile</b>	<b>11</b>
3.1 Planung und Analyse . . . . .	11
3.1.1 Vorüberlegungen . . . . .	11
3.1.2 Anforderungen . . . . .	12
3.2 Ergänzung von Landebahnen . . . . .	13
3.2.1 Datenergänzung . . . . .	13
3.2.2 Nutzerinteraktion . . . . .	15
3.3 Endanflug-Profile . . . . .	16
3.3.1 Direkter Anflug . . . . .	16
3.3.2 Trombone . . . . .	16
3.3.3 Point-Merge . . . . .	17
3.4 Ergänzung der Luftraumstrukturen . . . . .	18
<b>4 Implementierung</b>	<b>20</b>
4.1 Hinzufügen von Landebahnen . . . . .	20
4.1.1 Datenquelle . . . . .	20
4.1.2 Objektstruktur . . . . .	21
4.1.3 Bahnauswahl . . . . .	22
4.2 Benutzerinterface . . . . .	24

4.2.1	Tabellendarstellung Bahnen . . . . .	24
4.2.2	Parametrisierung der Endanflug-Profile . . . . .	25
4.3	Endanflug-Profile . . . . .	26
4.3.1	Abstrakte Basisklasse . . . . .	26
4.3.2	Profile . . . . .	27
<b>5</b>	<b>Beschreibung der Ergebnisse</b>	<b>32</b>
5.1	Vorstellung der Testdaten . . . . .	32
5.2	Was macht das Modul für Endanflug-Profile? . . . . .	32
5.2.1	Ergebnisse in Aviation Environmental Design Tool (AEDT) . . . . .	35
5.2.2	Ergebnisse in Air Traffic Optimization (AirTop) . . . . .	37
<b>6</b>	<b>Diskussion</b>	<b>39</b>
<b>7</b>	<b>Fazit und Ausblick</b>	<b>40</b>
	<b>Literaturverzeichnis</b>	<b>41</b>
	<b>Quellenverzeichnis</b>	<b>42</b>

# Abkürzungsverzeichnis

<b>AEDT</b>	Aviation Environmental Design Tool
<b>AirTOP</b>	Air Traffic Optimization
<b>DDR</b>	Demand Data Repository
<b>DFS</b>	Deutsche Flugsicherung
<b>DLR</b>	Deutsches Zentrum für Luft- und Raumfahrt e.V.
<b>FAA</b>	Federal Aviation Administration
<b>FAF</b>	Final Approach Fix
<b>FL</b>	Institut für Flugführung
<b>ICAO</b>	International Civil Aviation Organization
<b>IDE</b>	Integrated Development Environment
<b>LVS</b>	Luftverkehrssysteme
<b>RouGe</b>	Routengenerator
<b>RWY</b>	Runway
<b>STAR</b>	Standard Terminal Arrival Route
<b>TMA</b>	Flughafen-Nahbereich

# Abbildungsverzeichnis

2.1	Das Konzept von RouGe[3, S.8]	5
2.2	Visualisierung von Flugrouten	6
2.3	Trombone Anflug Frankfurt - Ausschnitt aus dem Arrival Chart für Ost-Anflug[6]	9
2.4	Darstellung der in der Echtzeitsimulation verwendeten Variante des „Point-Merge“-Verfahrens.[13]	10
3.1	Runway Objekte	13
3.2	Automatische Bahnauswahl	14
3.3	Programmablauf	15
3.4	Punkte der Trombone Struktur	17
3.5	Punkte der Point-Merge Struktur[7, vgl.]	17
3.6	Automatische Routenauswahl	18
4.1	Start- und Landebahn	21
4.2	Neue Objekte mit wichtigen Methoden	22
4.3	Auswahl der Namen für die Bahnen	23
4.4	Tabelle mit Bahninformationen	25
4.5	Tabelle mit Bahninformationen	26
4.6	Abstrakte Profil Klasse	26
4.7	Ablauf Profilerzeugung bei Direct Approach und Trombone Profil	27
4.8	Standard Werte und Vorgaben - Tombone	28
4.9	Trombone Anflug	28
4.10	Standard Werte und Vorgaben - Direct Approach	29
4.11	Direct Approach	29
4.12	Ablauf Profilerzeugung beim Point Merge Profil	30
4.13	Point-Merge Anflug	31
4.14	Standard Werte und Vorgaben - Point-Merge	31
5.1	Direct Approach Anflug	33
5.2	Trombone Anflug	34
5.3	Point-Merge Anflug	34
5.4	AEDT: Trombone Anflug	35
5.5	AEDT: Direct Approach	36
5.6	AEDT: Point-Merge Anflug	36
5.7	AirTOP: Frankfurt Süd Trombone-Anflug	37
5.8	AirTOP: München Northwest Point-Merge Anflug	38
5.9	AirTOP: München Nordost Point-Merge Anflug	38



# Vorwort

Während meiner Praxisphasen arbeite ich im Deutschen Zentrum für Luft- und Raumfahrt e.V. (DLR) im Institut für Flugführung (FL) in der Abteilung Luftverkehrssysteme (LVS). “Der Schwerpunkt der Forschungsarbeiten der Abteilung Luftverkehrssysteme liegt auf den Themenbereichen Modell- und Daten-basierte Leistungsbewertung, Leistungsbasiertes Flughafenmanagement und Fortschrittliche Luftverkehrsmanagementkonzepte. Neben der wissenschaftlichen Forschung, der methodischen Entwicklung und Umsetzung von operationellen Konzepten sowie der anwendungsnahen Validierung wird in der Abteilung an verbesserten Methoden zur Systemmodellierung, -analyse und -bewertung gearbeitet.”[10]

Die Erarbeitung der Simulationsmodelle für die verschiedenen Schnellzeitsimulatoren ist mit einem hohen Aufwand verbunden. Mit dem Routengenerator *RouGe* hat das Institut für Flugführung ein Projekt gestartet, das einen Großteil der Flugszenarien automatisiert erstellt. RouGe bekommt von verschiedenen Quellen Flugplandaten und modifiziert diese so, dass sie als Quelle für Schnellzeitsimulationen genutzt werden können.

Die Aufgaben dieser Bachelorarbeit werden im folgenden Bericht dargestellt.

# Kapitel 1

## Einleitung

Der Luftraum im Bereich moderner Flughäfen weist eine hohe Komplexität und Vielfalt an Flugrouten auf. Einen Teil davon bilden die Endanflüge auf die Landebahnen der Flughäfen. Um diese zu optimieren werden verschiedene Endanflug-Profile mit Hilfe von Schnellzeitsimulationen untersucht. So wurde zum Beispiel von der Deutschen Flugsicherung (DFS) eine Machbarkeitsstudie des Anflugverfahren *Point-Merge* für den Flughafen Frankfurt am Main simulatorgestützt durchgeführt.[7, S. 1] Die Modellierung und Einbindung verschiedener Profile ist mit einem hohem Aufwand verbunden, da die Anflugrouten ohne Automatisierung für jede Landebahn per Hand erstellt und in Szenarien eingefügt werden müssen. In dieser Arbeit soll ein Konzept entwickelt und implementiert werden, das eine flexible Generierung von Endanflug-Profilen ermöglicht. Dies umfasst die Ergänzung und den Ersatz von bestehenden Trajektorieninformationen um geeignete Endanflüge. Dadurch sollen aktuelle und mögliche zukünftige Endanflug-Profile erzeugt werden können.

### 1.1 Aufgabenstellung

Durch Literaturrecherche sollen die Eigenschaften verschiedener Anflug-Konzepte herausgearbeitet und verglichen werden. Darauf aufbauend ist ein Konzept zu erstellen und zu implementieren, welches die Erzeugung von Trajektorien-Abschnitten für verschiedene Anflugverfahren ermöglicht. Dabei sollen nicht nur existierende Anflugverfahren betrachtet werden, sondern es soll auch die Erstellung neuartiger Verfahren durch geeignete Parametrisierung möglich sein.

Es sollen im Rahmen der Arbeit exemplarisch zwei verschiedene Endanflug-Verfahren implementiert und mit Hilfe von Beispieldaten bewertet werden. Zur Bewertung kann beispielsweise das Aviation Environmental Design Tool (AEDT) verwendet werden. AEDT ist ein kommerzielles Werkzeug zur Bewertung von ATM-Systemen mit Fokus auf Lärm und Emissionen.

Die Umsetzung soll als Modul innerhalb von RouGe in der Programmiersprache Java erfolgen.

### 1.2 Hintergrund

Mit dem Softwarewerkzeug RouGe hat das Institut für Flugführung (FL) des Deutschen Zentrums für Luft- und Raumfahrt e.V. (DLR) ein Programm entwickelt, das realitätsnahe Flugszenarien weitgehend automatisiert aus Flugplandaten der europäischen Flugaufsichtsbehörde EUROCONTROL erstellen kann. Diese Daten heißen Demand Data Repository (DDR), und bestehen aus umfangreichen Trajektorien-Daten von realen Flügen, umfassen jedoch nicht den

Flughafen-Nahbereich (TMA). RouGe kann diese Daten bereits einlesen, verarbeiten und in verschiedenen Formaten für die Luftverkehrssimulation mit Schnellzeitsimulatoren exportieren. Ohne die Informationen aus dem TMA-Bereich sind diese Daten jedoch nur eingeschränkt nutzbar.

## 1.3 Motivation

Das Institut für Flugführung des Deutschen Zentrums für Luft- und Raumfahrt e.V. ist eine der weltweit führenden Forschungseinrichtungen für den Bereich Luftverkehrsmanagement. Dieser Bereich gewinnt immer mehr an Bedeutung, denn unter den internationalen Verkehrssektoren ist der Luftverkehrssektor einer der am stärksten wachsenden überhaupt. Dies zeigt sich zum Beispiel daran, dass viele der großen internationalen Flughäfen fast an ihren Kapazitätsgrenzen arbeiten. Das bedeutet einen enormen Strukturierungs- und Organisierungsaufwand der Flugsicherungen, Piloten und Fluglotsen. Im Nahbereich von Flughäfen müssen Piloten ganz auf die Anweisungen der Fluglotsen vertrauen, die sie sicher, schnell und kosteneffizient durch den Landeanflug leiten. Für die Entlastung der Fluglotsen und die Optimierung der Kapazitäten erforschen und entwickeln Forscher neue Endanflug-Profile.

Eine Schnellzeitsimulation besteht aus drei Prozessschritten. Dies sind die **Modellierung**, **Simulation** und **Auswertung**. Bei der Modellierung wird die Infrastruktur bestehend aus Flughafenstruktur und Luftraumstruktur angelegt, die Flugpläne erstellt (normalerweise für einen Tag) und verschiedene Verfahren eingebunden. Dies können zum Beispiel Anflug- oder Abflugverfahren sein. Die Schnellzeitsimulatoren führen dann mit den modellierten Daten automatisierte Simulationen durch. Je nach eingesetzter Simulationssoftware dauert eine Simulation zwischen wenigen Sekunden und wenigen Minuten. Die Ergebnisse werden anschließend analysiert und ausgewertet. Auswertungskriterien sind zum Beispiel Kapazitäten, Verspätungen oder Umweltparameter wie Treibstoffverbrauch und Emissionen.

Für die Entwicklung neuer Profile werden unter anderem Schnellzeitsimulatoren verwendet, die mit realen Flugplandaten arbeiten. Der Routengenerator (RouGe) erzeugt die verschiedenen Importdaten für die Simulatoren aus Flugplandaten der EUROCONTROL. Die Endanflug-Profile werden jedoch noch nicht von RouGe beachtet und müssen manuell modelliert werden. Für diese Aufgabe wird ein Modul benötigt, das Endanflüge flexibel generiert und automatisiert in die Flugplandaten integriert.

# Kapitel 2

## Grundlagen

Das folgende Kapitel soll die für das Verständnis der Arbeit benötigten Grundlagen vermitteln. Der Simulationsprozess und dessen Bedeutung für die Forschung wird neben den grundlegenden Elementen des Projektes erläutert. Zusätzlich werden die theoretischen Grundlagen zu den Endanflug-Profilen dargestellt.

### 2.1 Bedeutung der Simulation

Simulationen sind heutzutage eines der wichtigsten Hilfsmittel zur Analyse von komplexen Prozessen. Sie ersetzen reale Versuche, die zu teuer, zu zeitaufwändig oder zu gefährlich sind. Die Simulation ermöglicht es, ein Simulationsmodell unter unterschiedlichen Bedingungen zu betrachten und stetig zu verbessern. Der Simulationsprozess kann so oft wiederholt werden, bis die Ergebnisse eine optimierte realitätsnahe Form erreicht haben.

Die Möglichkeiten, die dieser Simulationsprozess bietet, eignen sich optimal zur Erforschung neuer und bestehender Koordinierungsprofile in der Flugführung. Dazu gehören unter anderem die verschiedenen Endanflug-Profile. Neue Verfahren können entwickelt und getestet werden, ohne echte Piloten und Flugzeuge einzusetzen und dadurch in Gefahr zu bringen.

Außerdem lassen sich verschiedene Parameter in die Simulation mit einbinden, die zum Beispiel Werte über den Kerosinverbrauch oder die Emission von Schadstoffen oder Lärm widerspiegeln. Die Simulationen und Auswertungen lassen sich anhand dieser Parameter und Flugplandaten größtenteils schon automatisiert durchführen.

Eines der Simulationstools, das im Institut für Flugführung eingesetzt wird, ist das von der Federal Aviation Administration (FAA) entwickelte AEDT. AEDT ist ein Auswertungswerkzeug, mit dem hauptsächlich Studien zur Untersuchung von Kerosinverbrauch, Lärmbelastung und Emissionen durchgeführt werden. Diese Punkte sind im Zusammenhang mit der Erprobung neuer Endanflugverfahren für einen Flughafen von besonderem Interesse, da man die Anwohner rund um einen Flughafen und die Fluggesellschaften so wenig wie möglich belasten und trotzdem eine optimale Ausnutzung der Kapazitäten eines Flughafens erreichen möchte.

Zur Auswertung der Ergebnisse kommt im Institut für Flugführung ein eigens entwickeltes Softwaresystem zum Einsatz. Dieses ermöglicht es den Nutzern, die Simulationsdaten unabhängig von den Simulationssystemen einheitlich zu bewerten. Die Ergebnisse werden für den Nutzer aufbereitet und zur Interpretation in verschiedenen Diagrammen zusammengefasst.

## 2.2 Technische Grundlagen

In diesem Abschnitt werden die technischen Grundlagen über die verwendeten Elemente und Hilfsmittel beschrieben.

### 2.2.1 Entwicklung

**Java** Für die Programmierarbeiten der Aufgaben wurde die Programmiersprache Java gewählt, da auch das Hauptprogramm in Java geschrieben ist. Java ist eine Programmiersprache und eine Laufzeitumgebung. Die zugrundeliegende Technologie dient als Basis für moderne Programme, wie Dienstprogramme, Spiele und Business-Anwendungen.[12]

**Eclipse** Eclipse ist das meistgenutzte Programmierwerkzeug zur Programmierung mit Java. Ein wichtiger Bestandteil von Eclipse ist die integrierte Entwicklungsumgebung (eng. Integrated Development Environment (IDE)) für Java. Eclipse ist eine quelloffene Programmierungsumgebung, um Programme in Java, C++ und vielen weiteren Programmiersprachen zu entwickeln.

### 2.2.2 Routengenerator (RouGe)

Der Routengenerator RouGe ist ein in Java implementiertes Programm, das spezialisierte Flugplandaten für verschiedene Schnellzeitsimulationen erzeugt. Diese Daten bekommt der RouGe von verschiedenen Quellen, wie zum Beispiel den DDR-Daten von Eurocontrol.

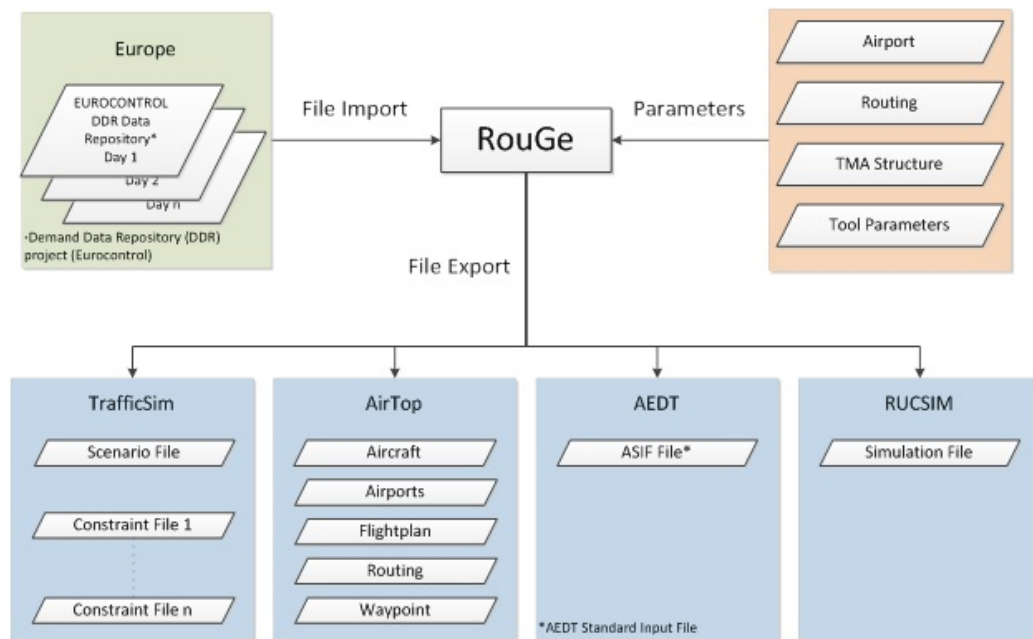


Abbildung 2.1: Das Konzept von RouGe[3, S.8]

Eurocontrol ist die europäische Aufsichtsbehörde für Luftfahrt, die an einem optimierten Luftverkehr in und um Europa arbeitet. Die DDR-Daten beinhalten alle Information zu den Flugrouten, Flugzeugen und angeflogenen Flughäfen. Die Daten liegen in unterschiedlicher Qualität und Quantität vor und sind oft redundant.“ In RouGe sollen diese Daten möglichst frei von Redundanzen, fehlerfrei und in homogenen Datenformaten zusammengeführt und anschließend durch gezielte Modifikationen an den Flugplänen verändert werden.”[1, S.17] In der

Abbildung 3 ist das Konzept von RouGe zu sehen. Der Nutzer lädt die Flugplandaten in RouGe und wird dabei von einem Dialog unterstützt, der die Erstellung von Schnellzeitsimulationsszenarien vereinfacht.

### 2.2.3 Auswertung

Für die Auswertung der Testdaten und die Funktionsweise des in dieser Arbeit entwickelten Konzepts werden das Visualisierungsmodul von RouGe und die Simulationsumgebung AEDT verwendet.

**Visualisierung in RouGe** RouGe besitzt ein Modul, das die fertig berechneten Flugrouten der Schnellzeitsimulationsszenarien auf einer Weltkarte darstellen kann. “Der Anwender soll dadurch noch vor dem Export für die Simulation dazu in der Lage sein, das Ergebnis der Routenerstellung zu untersuchen und gegebenenfalls anzupassen.”[1, S. 47]

Das Visualisierungsmodul verbindet die Wegpunkte einer Flugroute miteinander und zeichnet diese dann anhand der angegebenen geologischen Koordinaten an die richtige Position einer Weltkarte. Außerdem zeichnet es die Landebahnen ein, wenn diese als Parameter in den Szenariodaten vorhanden sind.

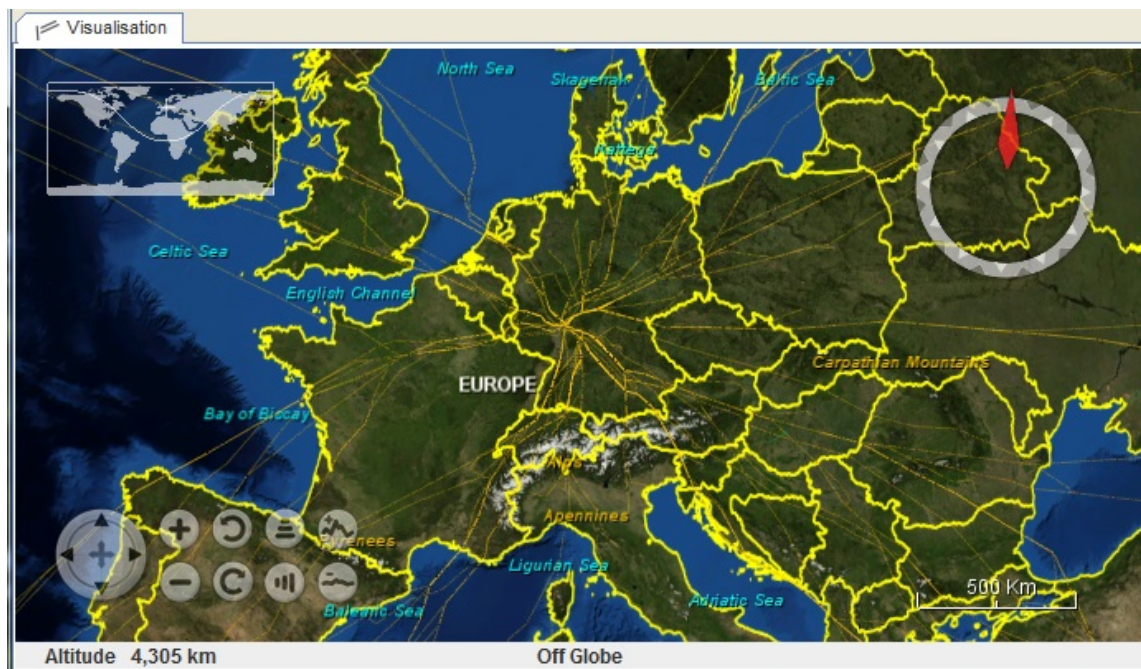


Abbildung 2.2: Visualisierung von Flugrouten

In Abbildung 2.2 kann man sieht man das Visualisierungsfenster von RouGe, das einen Europaabschnitt zeigt. Darauf werden Flugrouten aus einem Beispielszenario gezeigt, die vom Flughafen Frankfurt aus in alle Richtungen gehen.

**AEDT** RouGe besitzt ein Exportmodul, das die Szenariodaten zu einer für AEDT verständlichen Quelldatei aufbereiten kann. Aviation Environmental Design Tool (AEDT) ist ein von der Federal Aviation Administration (FAA) entwickeltes Auswertungswerkzeug, das den Emissionsausstoß und den Kerosinverbrauch berechnen kann, um Flugrouten zu optimieren. “Full

flight gate-to-gate analyses are possible for study sizes ranging from a single flight at an airport to scenarios at the regional, national, and global levels.”[2, S.1]

**AirTop** Ein weiteres RouGe-Exportmodul erzeugt spezifische Quelldaten für den Schnellzeitsimulator AirTop. AirTop steht für *Air Traffic Optimization*, ist eine auf Multiagenten basierende Schnellzeitsimulationssoftware und “deckt alle luftseitigen Prozesse von Gate zu Gate ab und ermöglicht nicht nur die Simulation einzelner Flughäfen sondern auch deren Verzahnung zum weltweiten Luftverkehr.”[3, S.3]

## 2.3 Luftraumstrukturen

In diesem Abschnitt werden die für diese Bachelorarbeit zum Verständnis benötigten Luftraumstrukturelemente vorgestellt.

**Wegpunkte** Ein Wegpunkt ist ein geographischer Punkt, an dem sich eine Route orientiert. Dieser ist mit geographischen Koordinaten und einem Namen versehen. Mit einer Abfolge von Wegpunkten werden Flugrouten und Luftstraßen beschrieben, denen dann Luftfahrzeuge folgen.

**Landeanflug** “Bezeichnung für den Flugabschnitt, der zwischen dem Reiseflug und der Landung liegt.” [4, S. 154f] Der Landeanflug wird in den meisten Fällen über standardisierte Routen geführt und durch Flugprofile beschrieben. Bei großen Flughäfen nutzen die Piloten meistens das Instrumentenanflugverfahren. Hierbei orientieren sie sich hauptsächlich an den Instrumenten und fliegen festgelegte Routen und Warterouten ab.

**Instrumentenanflugverfahren** Das Instrumentenanflugverfahren besteht aus fünf Segmenten, **Einflugstrecke (Arrival Route)**, **Anfangsanflug (Initial Approach)**, **Zwischenanflug (Intermediate Approach)**, **Endanflug (Final Approach)** und **Fehlanflug (Missed Approach)**. Auf der Einflugstrecke wird ein Flugzeug von seiner Reiseroute zu einem Fixpunkt geleitet, an dem der Landeanflug beginnt. Hierbei orientiert sich der Pilot an flughafenspezifischen Karten, welche die *Standard Terminal Arrival Route (STAR)* zeigen. Dies ist der *Initial Approach Fix*. Ab diesem Punkt beginnt der Anfangsanflug und das Flugzeug hat hohe Sinkraten und kann noch Warteschleifen durchfliegen, bis es den *Intermediate Approach Fix* erreicht. Für den Anfangsanflug benutzt der Pilot *Instrument Approach Charts*. Dann beginnt der Zwischenanflug, der dazu dient, das Flugzeug zu stabilisieren, bevor der Endanflug beginnt. Ab dem *Final Approach Fix* steuert das Flugzeug direkt auf die Landebahn zu. Wenn diese durch einen falschen Kurs verfehlt wird, muss der Pilot ab einem bestimmten Punkt den Fehlanflug einleiten, um hohen Hindernissen auf dem Boden auszuweichen und in die Anflugstrecke zurückzukehren. [5, vgl. S. 216] Auf den *Instrumental Approach Charts* sind die verschiedenen Warteverfahren und Endanflug-Profilen eingezeichnet, an die sich der Pilot halten muss, wenn er keine Anweisungen von den Lotsen bekommt.

**Start- und Landebahn** Start- und Landebahnen sind die Endpunkte einer Flugroute. Ihre Ausrichtung am Flughafen ist in den meisten Fällen gegen die Hauptwindrichtung gerichtet, so dass die Flugzeuge mit möglichst optimalen Windverhältnissen starten und landen können. Bei Gegenwind bekommt ein Flugzeug mehr Auftrieb und benötigt eine geringere Geschwindigkeit, um abzuheben oder zu landen. Die Enden einer Bahn werden als Bahnkopf bezeichnet.

Die Flugzeuge starten und landen aber einige Meter von diesen entfernt. Diese Punkte vor dem Bahnkopf auf der Bahn nennt man Schwelle. Die Schwellen sind die genauen Start- und Zielpunkte einer Flugroute.

## **2.4 Endanflugverfahren**

Große Flughäfen werden heutzutage von hunderten Flugzeugen pro Tag angesteuert. Durch die große Menge an Anflügen kommt es teilweise zu kapazitären Engpässen an einem Flughafen. Außerdem verursachen sie viel Lärm und einen hohen Schadstoffausstoß. Die Fluglotsen eines Flughafens haben die Aufgabe, die Flugzeuge über optimale Anflugrouten sicher auf den Flughafen zu leiten. Dabei müssen sie darauf achten, dass die Flugzeuge ausreichend Abstand zu einander halten und möglichst wenig in niedriger Höhe über bewohnte Gebiete fliegen. Zusätzlich müssen sie auch wirtschaftlich denken und die Flugzeuge möglichst schnell landen lassen.

Damit den Lotsen dies gelingt, werden verschiedene Verfahren entwickelt, welche die Anflugrouten der Flugzeuge beschreiben und wie diese optimal sortiert auf den Endanflug geleitet werden. Diese Verfahren sind die Endanflug-Profile, die in diesem Abschnitt erläutert werden.

### **2.4.1 Direct Approach**

Der Direct Approach beschreibt den direkten Anflug auf eine Landebahn. Hierbei muss der Flieger keine Warterouten abfliegen. Dieses Verfahren wird angewendet, wenn es nur wenig Anflüge auf einen Flughafen gibt, sodass der Fluglotse die Kapazität einer Landebahn nicht beachten muss. Der direkte Anflug beschreibt den kürzesten Weg von der Reiseroute zum Finalen Anflug Punkt, auch Final Approach Fix (FAF) genannt. Dieser Punkt ist ein fixer Punkt, an dem jedes Flugzeug vorbeikommen muss und den Endanflug auf die Landebahn beginnt.



## 2.4.2 Trombone Verfahren

Der Begriff Trombone stammt aus dem englischen und heißt auf deutsch *Posaune*. Dieses Verfahren wurde nach dem Musikinstrument benannt, da es Ähnlichkeiten mit dessen Aussehen und Funktionsweise besitzt. Es wird dazu genutzt, die Flugzeuge optimal zu staffeln und zum richtigen Zeitpunkt landen zu lassen. Hierbei fliegen die Flugzeuge parallel zur Endanflugroute, bis sie ungefähr auf der Höhe der Landebahn sind. Dann fliegen sie eine 180 Grad-Kurve und bewegen sich auf einer Parallelroute zwischen der Endanflugroute und dem vorherigen Weg. Um zum Final Approach Fix zu kommen müssen sie noch eine 180 Grad Kurve in Richtung Landebahn fliegen, wenn der Lotse die Anordnung dafür gibt. Hier setzt die Staffelung mit ein. Der Lotse bestimmt den Zeitpunkt, wann der Pilot auf den Endanflug eindreht. Dieses Verfahren ist auf Abbildung 2.3 zu erkennen.

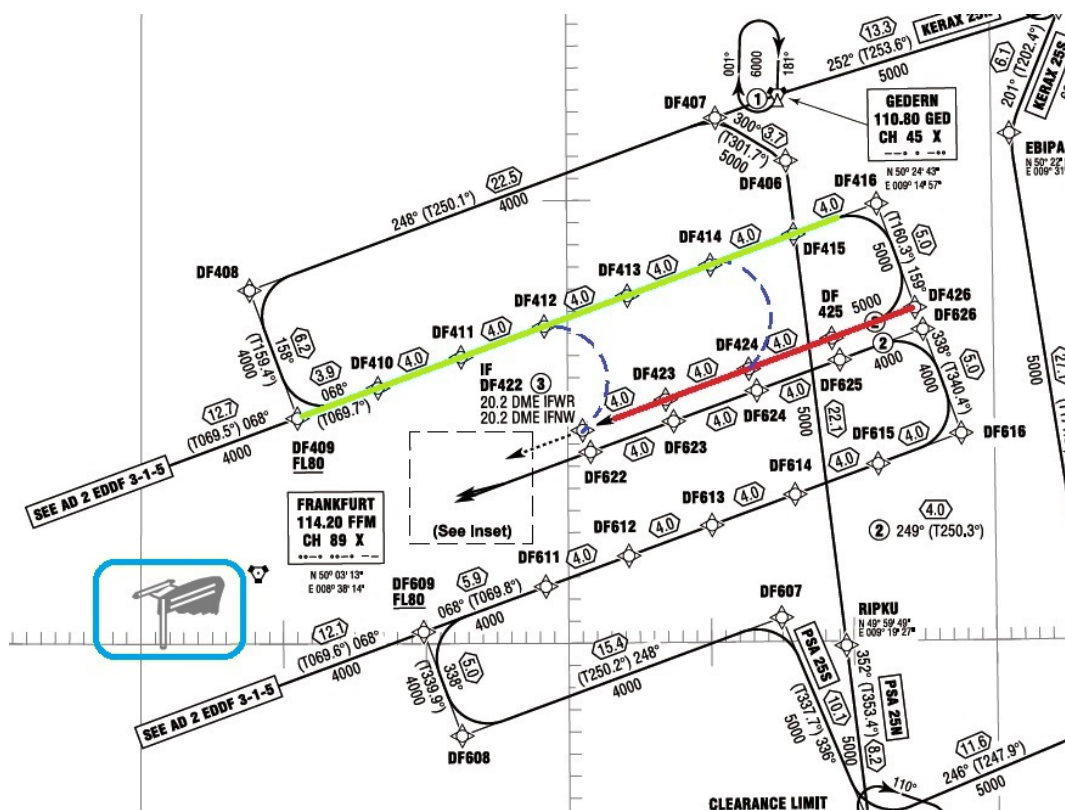


Abbildung 2.3: Trombone Anflug Frankfurt - Ausschnitt aus dem Arrival Chart für Ost-Anflug[6]

Diese Abbildung zeigt einen Ausschnitt aus der Arrival Chart für Ostanflüge auf den Flughafen Frankfurt aus dem Luftfahrthandbuch Deutschland der Deutschen Flugsicherung[6]. Zu sehen ist die Grundstruktur des Anflugverfahrens mit Flughafen (blau markiert), der Wartestrecke, die sich vom Flughafen entfernt (grün markiert), der finalen Anflugstrecke (rot markiert) und beispielhaften Eindrehwegen (lila und gestrichelt markiert). Solange der Lotse kein Signal gibt, entfernt sich das Flugzeug auf der Wartestrecke weiter vom Flughafen. Wenn der Lotse dem Piloten das Okay für den Endanflug gibt, dreht dieser auf die finale Anflugroute ein. Die Funktion des Zuges einer Posaune entsteht dadurch, dass die Flugzeuge an unterschiedlichen Punkten eindrehen, um den optimalen Abstand zueinander zu bekommen, der je nach Situation unterschiedlich groß ausfallen kann.

### 2.4.3 Point-Merge Verfahren

Das Point-Merge Verfahren ist eine innovative Möglichkeit die Flugzeuge lärmoptimiert zu staffeln. "Point Merge is a systemised method for sequencing arrival flows developed by the EUROCONTROL Experimental Centre in 2006." [14] "Bei 'Point Merge' werden Anflüge in relativ großer Höhe kanalisiert und mit dem notwendigen Sicherheitsabstand kontinuierlich sinkend zum Endanflug geführt." [7] Je nachdem aus welcher Richtung die Flugzeuge ankommen, werden sie auf zwei verschiedene *Sequencing Legs* geleitet. Dies sind parallel entgegengesetzt laufende Routen, die einem Kreisausschnitt folgen. Sie verlaufen meist in einem Abstand von zwei nautischen Meilen zueinander und sind zusätzlich vertikal separiert, damit sich die Flugzeuge nicht queren. Die Flugzeuge fliegen dann auf Anweisung der Lotsen zum Kreismittelpunkt, dem Merge-Punkt. Die Flugzeuge fliegen dabei nicht über den Kreisausschnitt hinaus, der von den beiden Einflugpunkten begrenzt wird. Am Ende des Ausschnittes drehen die Flugzeuge automatisch zum Merge-Punkt ein. Von da aus fliegen sie zur Anfluggrundlinie und beginnen mit dem Endanflug.

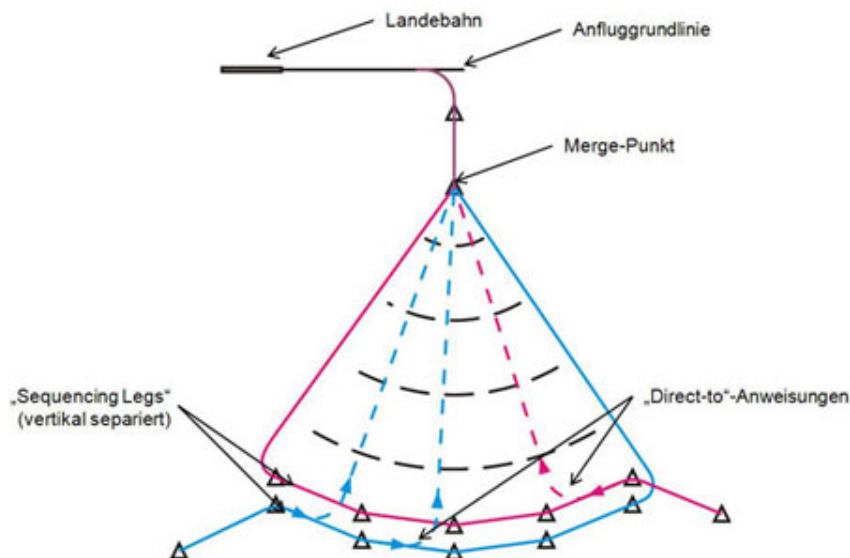


Abbildung 2.4: Darstellung der in der Echtzeitsimulation verwendeten Variante des „Point-Merge“-Verfahrens. [13]

Die Staffelung der Flugzeuge lässt sich mit dieser Struktur einfach planen, da die Flugzeuge alle gleich lange von den *Sequencing Legs* zum Merge-Punkt brauchen, während sich die Flugzeuge beim Trombone-Anflug immer weiter vom Flughafen entfernen. Die Arbeitslast der Lotsen wird auch dadurch verringert, dass diese nur noch *Direct-to-Anweisungen* geben müssen, woraufhin die Flugzeuge zum Merge-Punkt fliegen. Die dafür benötigten Kursinformationen bekommen die Piloten aus den Anflugkarten des jeweiligen Flughafens. Dieses Verfahren wird zur Zeit in Oslo (seit 2011), Dublin (2012), Seoul (2012) und Paris (2013) angewendet. Im Jahr 2015 werden London Gatwick und London City hinzukommen. [14]

# Kapitel 3

## Konzept flexibler Endanflug-Profile

Mit dem Routengenerator RouGe hat das Institut für Flugführung ein Projekt ins Leben gerufen, das die einheitlichen Flugplandaten von EUROCONTROL für die verschiedenen Simulationstools aufbereiten und mit zusätzlichen Parametern versehen kann. Da die Daten von EUROCONTROL als Start und Ziel nur den jeweiligen Flughafen angeben, starteten und endeten die Flüge immer abrupt. Mit diesen Eigenschaften lassen sich Simulationen ohne aufwändiges manuelles Eingreifen nur außerhalb des TMA-Bereiches ausführen.

Damit die Simulation der Landeanflüge im Flughafennahbereich durch RouGe-Daten möglich wird, wird mit dieser Bachelorarbeit ein Konzept entwickelt, das eine flexible Generierung von Endanflug-Profilen ermöglicht.

### 3.1 Planung und Analyse

In den folgenden Abschnitten werden die Vorüberlegungen behandelt, die nötig sind, damit die neuen Funktionalitäten in RouGe flexibel und robust implementiert werden können. Dabei wird auf die Planung der Arbeit und die Analyse der Basis und der Gegebenheiten genauer eingegangen.

#### 3.1.1 Vorüberlegungen

Im Rahmen dieser Bachelorarbeit soll RouGe um die Möglichkeit erweitert werden, die Flugrouten um flexibel generierte Endanflug-Profile zu ergänzen oder zu ersetzen. Dies ermöglicht die einfache Zusammenstellung von Szenarios für Simulationen im Nahbereich von Flughäfen. Bisher verarbeitet RouGe die Flugrouten so, wie sie in den Demand Data Repository (DDR) Quelldaten beschrieben sind. Wenn die Quelldaten eine geringere Auflösung haben, besitzt eine Flugroute keine genauere Struktur im Endanflug. Außerdem starten und landen alle Flüge auf den Referenzkoordinaten eines Flughafens und nicht auf den Start- und Landebahnen. Dies ist aber nur ein einziger Punkt, der die Position des Flughafens bestimmt. Die Flugrouten müssen also auch um Informationen über Start- und Landebahnen ergänzt werden.

Nachdem ein Szenario in RouGe um Start- und Landebahnen ergänzt wurde, müssen der Start- und der Landepunkt eines Flugzeuges festgelegt werden. Auf einem Flughafen sind dies die Bahnschwellen, von denen ein Flugzeug aus abhebt und landet. Es muss ein Algorithmus implementiert werden, der die richtige Bahn für die Flugzeuge auswählt. Der Nutzer legt dafür fest, von welchem Typ eine Bahn ist, ob Flugzeuge nur landen oder starten dürfen, oder ob beides möglich ist.

Anschließend soll der Nutzer die Endanflug-Profile aus einer Auswahl implementierter Profile definieren. Damit die Profile von RouGe verarbeitet werden können und auch neue Anflugprofile implementiert werden können, brauchen diese im Programm eine abstrakte Basisklasse, welche die Struktur der Klassen für die verschiedenen möglichen Profile definiert.

Für den Informationsfluss zwischen dem Nutzer und RouGe werden auch neue Dialoge benötigt, in denen die Auswahl und Definition gesteuert wird. Damit die Arbeitslast für den Anwender nicht zu groß ist, müssen eine Filterung der Daten und Speicherung der Änderungen stattfinden. Diese Maßnahmen schützen vor redundanten Eingaben und langen Suchprozeduren.

Die Ergänzung der Szenariodaten muss auf der bestehenden Struktur aufbauen, sodass die anderen Module von RouGe ohne Modifikation weiterhin funktionieren. Die Programmklassen der Endanflug-Profile müssen deshalb Anflugrouten bestehend aus Wegpunkten generieren, mit denen die Flugrouten ergänzt werden.

Nachdem die Szenarien die neuen Daten erhalten haben, müssen diese auch noch im Visualisierungsmodul von RouGe dargestellt werden. Die Anflugrouten sind in den Flugrouten schon vorhanden und benötigen keinen zusätzlichen Aufwand. Die Start- und Landebahnen müssen jedoch gezeichnet und eingebunden werden.

### 3.1.2 Anforderungen

Einer der ersten Schritte der Softwareentwicklung ist die Aufstellung von Anforderungen, damit das Ergebnis am Ende den gestellten Erwartungen entspricht. Beim Aufstellen der Anforderungen wird schon darauf geachtet, dass wichtige Qualitäts- und Sicherheitskriterien eingehalten werden. Richtig gestellte Anforderungen sind der erste Schritt auf dem Weg zu einer hochwertigen Software. Im folgenden Abschnitt wird näher auf die in dieser Bachelorarbeit gestellten Anforderungen eingegangen.

**Anforderungen im Rahmen der Bachelorarbeit** Es soll ein RouGe-Modul entwickelt werden, das die Erzeugung von Trajektorien-Abschnitten für verschiedene Anflugverfahren ermöglicht. Es sollen neben existierenden Anflugverfahren auch neuartige Verfahren durch geeignete Parametrisierung betrachtet werden können. Zwei Endanflug-Verfahren sollen hierfür exemplarisch implementiert und anhand von Beispieldaten bewertet werden.

**Datenhaltung und Programmablauf** Das Modul muss so entwickelt werden, dass es den bestehenden Programmablauf um weitere Funktionen ergänzt und nicht ersetzt und dass bei Nichtverwenden der Programmablauf unverändert bleibt. Des Weiteren soll die bestehende Datenstruktur so weit wie möglich verwendet werden, um Redundanzen zu vermeiden.

Durch einen strukturellen und logischen Aufbau der neu entstehenden Benutzeroberflächen soll der Anwender leicht durch das neue Modul finden und Hilfestellungen erhalten. Des Weiteren sollen im Programmablauf Erleichterungen für den Anwender entstehen, wie das Speichern und Wiederverwenden von modifizierten Parametern und das Vorfiltern von Elementen, sodass nur sinnvolle Daten zur Bearbeitung zu Verfügung stehen.

**Ergänzung der Luftraumstruktur** Die vorhandenen Luftraumstrukturen der Szenarien sollen mit bereits vorhandenen Mitteln erweitert, ergänzt und ersetzt werden. Die Flugrouten sollen um weitere Wegpunkte vom gleichen Typ wie die Vorhandenen ergänzt werden.

**Visualisierung** Zur Überprüfung der Ergebnisse soll das Visualisierungsmodul von RouGe so angepasst werden, dass es die Ergänzungen direkt anzeigt. Außerdem sollen die Start- und Landebahnen ergänzt und verwendet werden, wenn diese in den Szenariodaten vorhanden sind.

## 3.2 Ergänzung von Landebahnen

Eine sinnvolle Simulation im Flughafennahbereich mit verschiedenen Endanflug-Profilen ist nur möglich, wenn die Start- und Landebahnen mit berücksichtigt werden. Neben den Bahnen benötigen die Flugrouten auch Informationen, wie die Bahnen mit einbezogen werden. Im folgenden Kapitel wird das Konzept zur Ergänzung der Landebahnen erläutert.

### 3.2.1 Datenergänzung

RouGe und die DDR-Daten von EUROCONTROL beinhalten noch keine Informationen über Start- und Landebahnen. Diese müssen also auch noch hinzugefügt werden. Als Informationsquelle über die Runway (RWY) (englisch für Start- und Landebahn) steht eine Tabelle[17] mit den aktuellen Informationen über die meisten RWYs, die es auf der Welt gibt, zur Verfügung. Auf Grundlage dieser Informationsquelle kann ein Importmodul entwickelt werden, das die Informationen aus dem Dokument ausliest, verarbeitet und in eine geeignete Datenstruktur speichert. Diese Struktur muss in RouGe neu implementiert werden, da noch keine geeignete Struktur vorhanden ist.

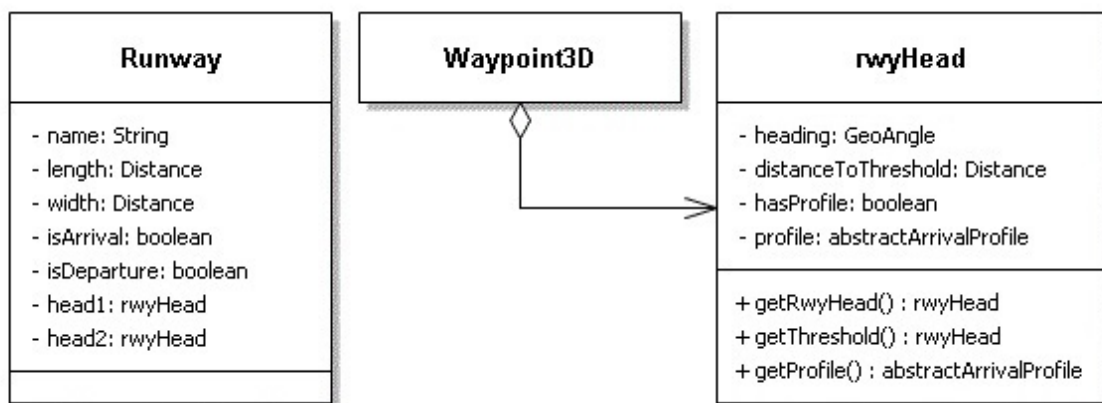


Abbildung 3.1: Runway Objekte

Die RWYs benötigen ein Objekt, das die Informationen über den Namen der RWY, dessen Dimensionen, der Art und der Bahnköpfe beinhaltet. Für die Bahnköpfe wird ein eigenes Objekt benötigt. Dieses stellt die Informationen für die Bahnköpfe und die Bahnschwellen zur Verfügung. Es beinhaltet die Informationen über dessen Lage als dreidimensionaler Wegpunkt und Richtung, sowie die Informationen über das Endanflug-Profil. Die Richtung wird als geographischer Winkel oder auch Kurswinkel gespeichert. Außerdem benötigt es die Distanz zwischen Bahnende und Schwelle, um Letztere zu Berechnen.

Bei der Berechnung der Flugrouten kann der Anwender über einen Flughafenfilter angeben, welche Flughäfen berücksichtigt werden. Diese Eigenschaft kann bei der Datenergänzung dazu verwendet werden, die Start- und Landebahnen vorzufiltern. Dazu wird aus den Informationen der Filtereingabe eine Liste mit Flughäfen erstellt, dessen Start- und Landebahnen relevant sind.

Beim Einlesen der Datenquelle müssen nun nur noch die Daten eingelesen werden, die relevant sind. Dies erspart dem Nutzer das Durchsuchen von circa 40.000 RWY nach denen, die er benötigt.

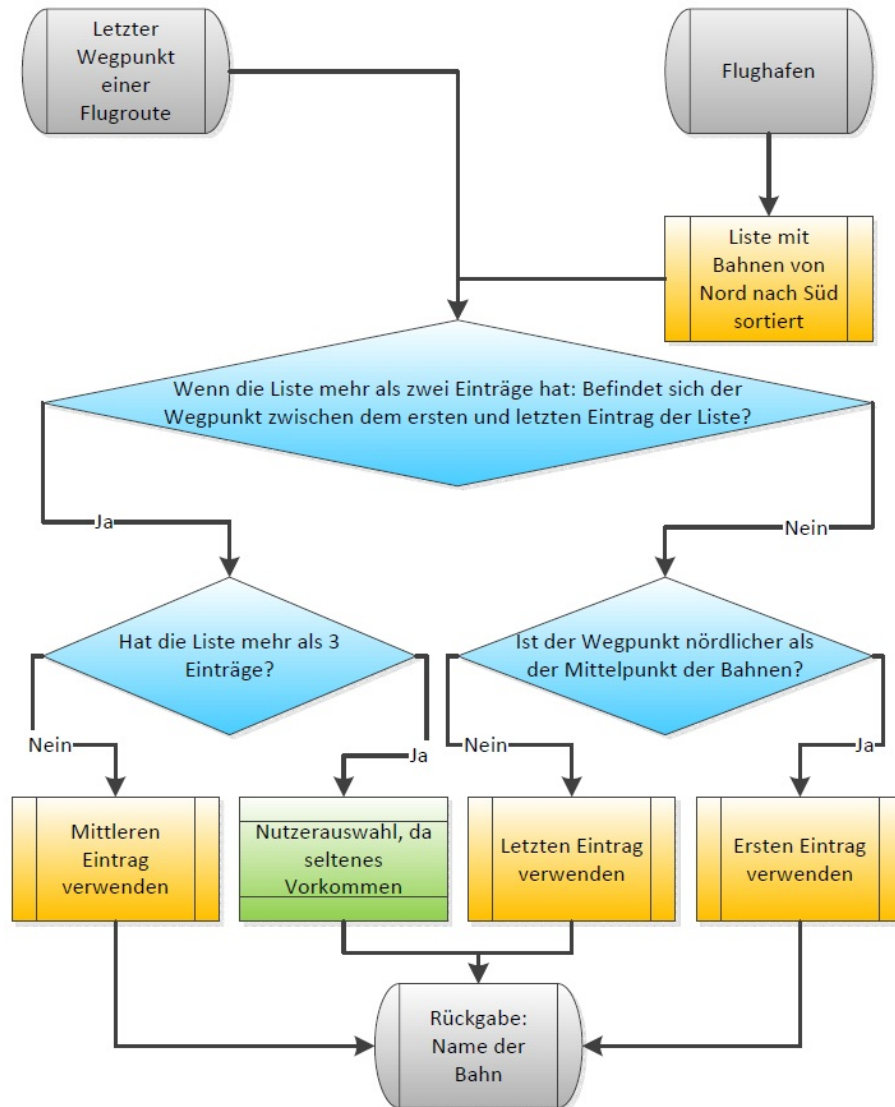


Abbildung 3.2: Automatische Bahnauswahl

Zusätzlich muss der Anwender noch angeben, von welchem Typ eine RWY ist. Hierbei legt er fest, ob die Bahn eine Startbahn, Landebahn oder beides ist. Nun soll der Nutzer die benötigten RWYs auswählen. Aus den ausgewählten Bahnen wird dann basierend auf der Anflugrichtung automatisch die richtige Bahn für jeden Flug ausgewählt.

Eine automatische Auswahl muss gestartet werden, wenn ein Flughafen mehr als eine Bahn eines Typs hat, die vom Nutzer ausgewählt wurden. In der Abbildung 3.2 erkennt man den dazugehörigen Ablauf in Form eines Flussdiagramms.

Die zusätzlichen Angaben, die der Nutzer angegeben hat, werden abgespeichert und für den nächsten Auswahlprozess geladen, damit er weniger Aufwand hat, die Informationen zu ergänzen.



### 3.2.2 Nutzerinteraktion

Das Modul soll in RouGe nur gestartet werden, wenn der Nutzer dies ausdrücklich wünscht. Dazu muss eine Auswahlbox in den Flugberechnungsdialog eingebaut werden, die das Modul starten lässt. In einem neuen Dialogfenster soll das Quelldokument mit den Informationen über die RWY vom Anwender eingelesen werden. Die gefilterten Daten werden dann in einer Tabelle angezeigt. Diese Tabelle hat zwei zusätzliche Spalten, in denen der Nutzer pro Zeile ein Auswahlménü erhält, welche die Auswahl zusätzlicher Daten, wie dem Bahntyp, ermöglichen. Nach der Bearbeitung der Tabelle werden die benötigten Bahnen ausgewählt.

In einem weiteren Dialog fügt der Nutzer den Landebahnen Endanflug-Profil hinzu und definiert die Parameter hierfür. Es soll eine Auswahl an bereits implementierten Profilen zur Verfügung stehen.

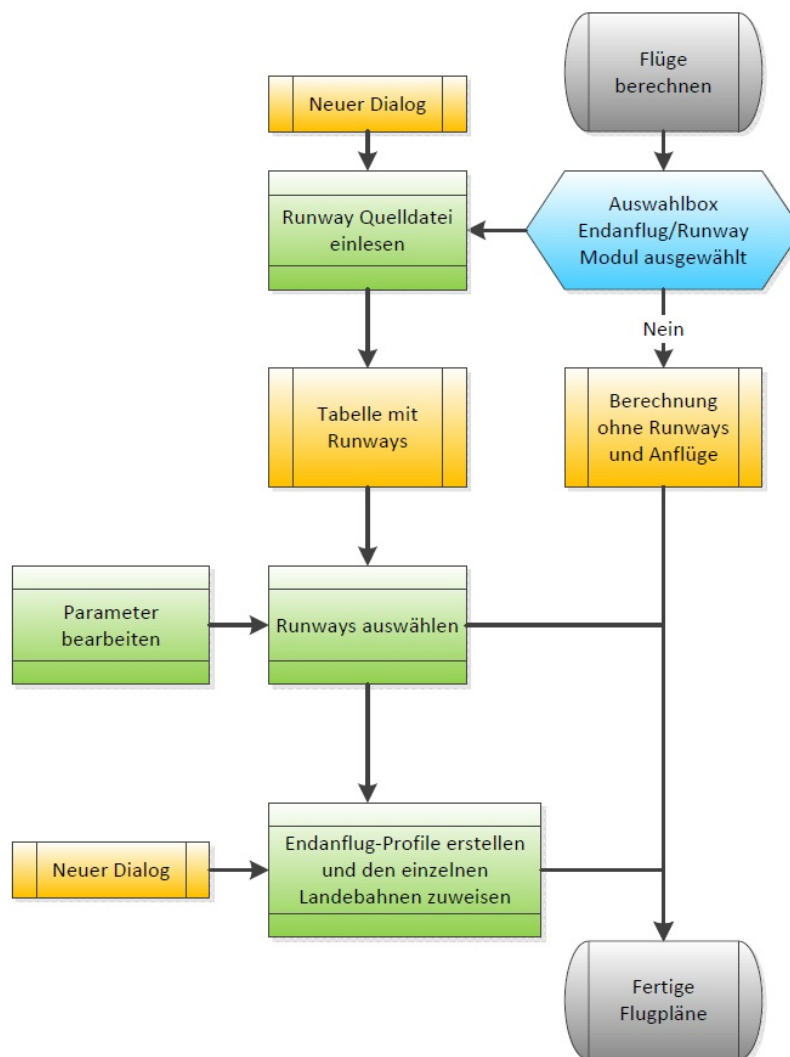


Abbildung 3.3: Programmablauf

### 3.3 Endanflug-Profile

Nachdem die Landebahnen verfügbar sind, können ihnen verschiedene Endanflug-Profile zugewiesen werden. Die Endanflug-Profile sollen sich alle auf die gleiche Art einstellen lassen und von RouGe gleich behandelt werden. Außerdem sollen vom Anwender möglichst wenige Parameter verlangt werden. Dies lässt sich mit einer abstrakten Klasse umsetzen, die Methoden für die abgeleiteten Klassen vorgibt. “Sie kann dann als Modellierungsklasse eine Ist-eine-Art-von-Beziehung ausdrücken und Signaturen für die Unterklassen vorgeben. Eine Oberklasse besitzt dabei Vorgaben für die Unterklasse, das heißt, alle Unterklassen erben die Methoden.”[9, Kapitel 5.12] Die einzelnen Endanflug-Profile sollen als Unterklasse dieser Modellierungsklasse beschrieben werden, was dazu führt, dass man den Landebahnen Objekte zuweisen kann, die durch die Unterklassen definiert sind. Die Objekte sind alle vom gleichen Typ, da sie durch die abstrakte Modellierungsklasse beschrieben werden. RouGe kann die Profile dann mit den selben Funktionen verarbeiten.

In der Modellierungsklasse muss eine Methode vorgegeben werden, mit welcher der Anflugweg zurückgegeben wird. Dieser Methode muss beim Aufruf noch der letzte Wegpunkt einer Route übergeben werden, damit eine individuelle Anflugroute für einen Flug zwischen dem letzten Wegpunkt und der Landebahn berechnet werden kann.

Dem Anwender soll für jede Landebahn eine Tabelle pro Endanflug-Profil zur Verfügung stehen, in denen er die spezifischen Parameter eingeben kann. Diese Tabellen haben den gleichen Aufbau. In einer Zeile wird eine Koordinate der Anflugstruktur beschrieben. Der Punkt kann entweder durch Längen- und Breitengrad definiert sein, oder durch Entfernung und geographischen Winkel zum nächstgelegenen Punkt in Richtung Landebahn. In der ersten Zeile soll der erste Punkt vor der Landebahn definiert werden. In den darauffolgenden Zeilen einer Tabelle sollen immer die nachfolgenden Punkte definiert werden. Neben den Ortsangaben bekommt jeder Wegpunkt einen Namen und eine Höhenangabe.

Für die verschiedenen Endanflug-Profile gibt es einen unterschiedlichen Bedarf an Fixpunkten. Für jeden benötigten Fixpunkt gibt es eine vorgefertigte Zeile in der Tabelle. Bis auf den geographischen Winkel sind hier Standardwerte vorgegeben, die geändert werden können. Weitere Zeilen sollen vom Nutzer hinzugefügt werden können.

#### 3.3.1 Direkter Anflug

Beim direkten Anflug geht es darum, dass das Flugzeug auf dem einfachsten Weg von seiner Reiseroute auf die richtige Landbahn gelangt. Dafür wird mindestens ein Fixpunkt benötigt. Dies ist der *Final Approach Fix*, ab dem das Flugzeug ohne weitere Kursänderungen direkt auf die Landebahn zusteuert.

Der Anwender kann diesem Punkt noch weitere hinzufügen, die Flugzeuge im Landeanflug alle nacheinander abfliegen sollen. Die Profilklassse muss aus den in der Tabelle vorhandenen Einträgen eine Route aufbauen. Aus den einzelnen Einträgen werden Wegpunkte erstellt und in einer Liste zusammengefasst.

#### 3.3.2 Trombone

Um die Trombone-Struktur zu erzeugen, wird der Weg benötigt, den ein Flugzeug mindestens fliegen muss. Dieser besteht aus mindestens vier Eckpunkten, welche die Struktur definieren. Das sind der *Final Approach Fix* (A), der *Initial Approach Fix* (D) und zwei Punkte (B und C), an denen jeweils eine 90 Grad Kurve geflogen wird, wie es auf Abbildung 3.4 zu sehen ist. Diese



vier Punkte muss der Anwender beschreiben und kann ihnen noch weitere hinzufügen, um die Struktur zu vergrößern. Die Profilklassse kann die Punkte auf die gleiche Art verarbeiten wie beim direkten Anflug. Die durch die posaunenartige Erweiterung entstandenen Verschiebungen der Punkte B und C berechnen die Simulatoren selbstständig (B1/B2 und C1/C2). Dazu wird lediglich die kürzeste Route benötigt.

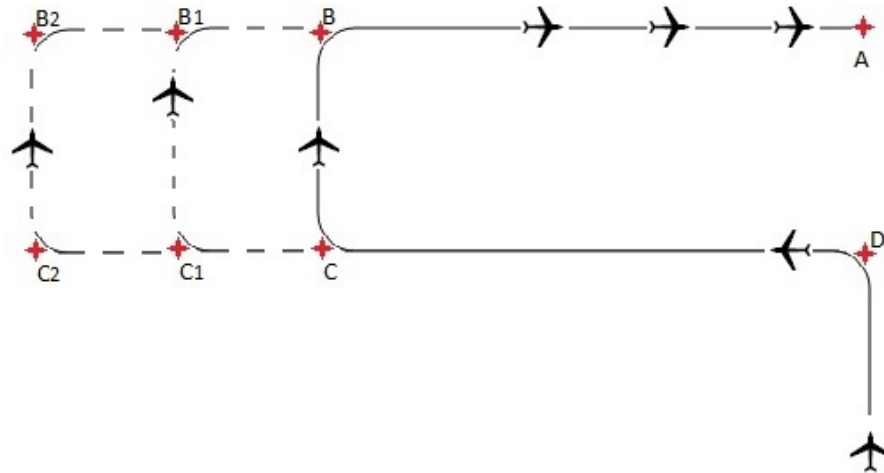


Abbildung 3.4: Punkte der Trombone Struktur

### 3.3.3 Point-Merge

Für die Point-Merge Struktur werden genau vier Fixpunkte benötigt, mit dessen Hilfe die verschiedenen Wege der Flugzeuge auf der Struktur berechnet werden können, wie auf Abbildung 3.5 zu sehen. Die benötigten Fixpunkte sind die beiden Einflugpunkte (Punkte A und B), der Merge-Punkt (M) und der *Final Approach Fix* (nicht auf Grafik zu sehen). Für diese Struktur müssen die geographischen Winkel und die Entfernungen sowie die Höhe angegeben werden. Die beiden Einflugpunkte beziehen sich mit der Entfernung auf den Merge-Punkt.

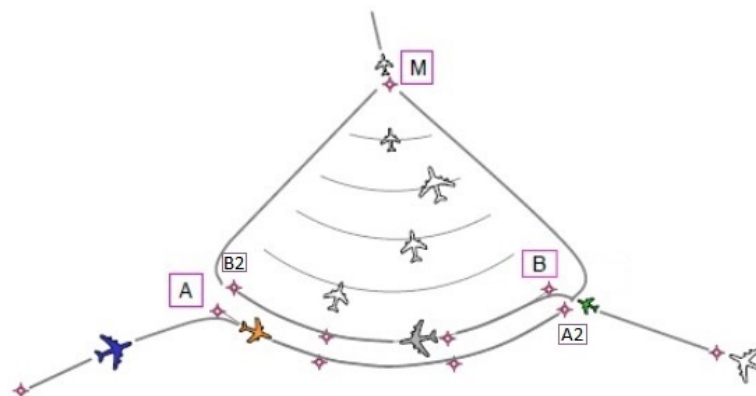


Abbildung 3.5: Punkte der Point-Merge Struktur[7, vgl.]

In der Unterklasse zu diesem Profil müssen dann zwei weitere Punkte (A2 und B2) berechnet werden, da ab den Einflugpunkten die beiden Sequencing Legs starten und zwei parallel verlaufende Kreissegmente bilden. Die beiden Kreissegmente sind zusätzlich vertikal separiert, damit sich die Flugzeuge in keinem Fall auf Kollisionskurs befinden. Die berechneten Punkte

sind die Endpunkte dieser Segmente. Der Endpunkt vom ersten Einflugpunkt hat dessen Höhe und Entfernung zum Merge-Punkt, aber den Winkel des zweiten Einflugpunktes. Bei dem anderen Endpunkt ist es genau andersherum. Zwischen den Einflug- und Endpunkten sollen anschließend Bögen geflogen werden, die an jedem Punkt die gleiche Entfernung zum Merge-Punkt haben. Dazu wird der Winkel zwischen den Eckpunkten in zehn gleichgroße Teile geteilt. Die Teilwinkel werden auf den Winkel des ersten Einflugpunktes addiert. Zusammen mit den beiden Entfernungsangaben entstehen zwei Routen, die einem Kreisabschnitt ähneln. Von den Endpunkten aus gehen beide Routen auf den Merge-Punkt zu und werden zu einer Route zusammengeführt, die über den *Final Approach Fix* zur Landebahn geführt wird.

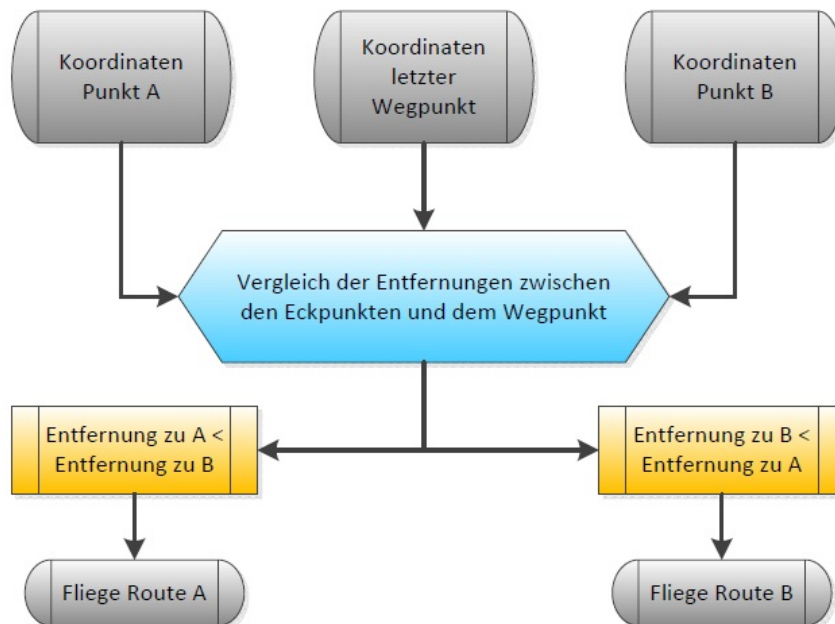


Abbildung 3.6: Automatische Routenauswahl

Da durch diese Struktur mehr als eine Möglichkeit gegeben wird, muss ein Algorithmus entscheiden, welcher Flug welche Route nimmt. Dazu wird die Koordinate des letzten Wegpunktes einer Flugroute benötigt. Im Algorithmus wird die Entfernung dieses Punktes zu den Einflugpunkten verglichen. Es wird dann die Route ausgewählt, deren Einflugpunkt dichter an der Flugroute liegt.

### 3.4 Ergänzung der Luftraumstrukturen

Nachdem der Nutzer alle Eingaben getätigt hat, stehen nun Flughäfen mit Landebahnen zur Verfügung, deren Landeswellen teilweise Endanflug-Profilen zugewiesen sind. Diese Daten sollen bei der Routengenerierung berücksichtigt werden. Dabei wird für jeden Flug die Route aus den Wegpunktdaten neu erstellt und um die ergänzten Daten erweitert. Dazu soll überprüft werden, ob ein Flughafen neue Strukturen besitzt. Wenn dies der Fall ist, wird die richtige Anflugroute zurückgegeben. Hierfür muss der Flug und der letzte Wegpunkt einer Route der Methode für die Anflugrouten-Rückgabe übergeben werden. Aus den Daten des Fluges wird die richtige Landebahn gewählt. Wenn diese ein Endanflug-Profil hat, wird die Anflugroute mithilfe des letzten Wegpunktes berechnet, wie in Kapitel 3.2.1 beschrieben. Andernfalls wird nur die Koordinate der Landeschwelle als Wegpunkt zurückgegeben, damit die Flüge ein Ziel haben.

Die Visualisierung von RouGe soll die fertigen Flugrouten mit Anflugverfahren anzeigen und auch die Start- und Landebahnen zeichnen. Da die Routen komplett aus Wegpunkten bestehen, muss die bestehende Visualisierung hierfür nicht angepasst werden. Für die Start- und Landebahnen müssen neue Zeichenobjekte generiert werden, die auch die richtigen Informationen erhalten.

# Kapitel 4

## Implementierung

Dieses Kapitel soll die programmiertechnische Umsetzung des erstellten Konzepts erläutern sowie die Funktionen und Programmabläufe beschreiben.

Mit dieser Arbeit wird RouGe um zwei zusammenarbeitende Module erweitert. Eines dieser Module erweitert die Flugplandaten um detaillierte Informationen über Start- und Landebahnen und weist diese den Flugrouten zu. Das zweite Modul kann dann mit den neuen Informationen Anflugrouten erzeugen. Für diese stehen verschiedene Endanflug-Profile zur Verfügung, die der Nutzer mit Informationen füllen kann.

### 4.1 Hinzufügen von Landebahnen

In diesem Abschnitt wird genauer auf das Ergänzen der Flugplandaten um Informationen über Start- und Landebahnen eingegangen.

Die benötigten Informationen stehen als Tabelle [17] mit den aktuellen Daten über viele bedeutsame Bahnen der Welt zur Verfügung. Für diese neuen Informationen werden Objekte erzeugt, in welche die Daten gespeichert werden. Sobald die Start- und Landebahnen zur Verfügung stehen, können diese den Flugrouten automatisch zugewiesen werden.

#### 4.1.1 Datenquelle

Als Datenquelle steht eine Tabelle zur Verfügung, welche die Informationen über Start- und Landebahnen beinhaltet. Die Daten sind nach den ICAO-Codes der Flughäfen sortiert und beinhalten die Informationen über dessen Dimensionen und den beiden Bahnrichtungen. Ein ICAO-Code ist eine Abkürzung für die eindeutige Identifikation eines Flughafens oder anderer Objekte der Luftfahrt. Diese Abkürzungen sind ein von der International Civil Aviation Organization (ICAO) entwickelter Standard. Die ICAO ist eine Luftfahrtorganisation aus dem Bereich Forschung. [15, vgl.]

Jede Bahnrichtung hat als Anfangspunkt einen definierten Bahnkopf, für den die dreidimensionalen Koordinaten bestehend aus Längengrad, Breitengrad und Höhe existieren. Eine Bahnrichtung hat auch eine Identifikation, einen geographischen Winkel und Informationen über die Lage einer Bahnschwelle, wenn deren Position von dem Bahnkopf abweicht.

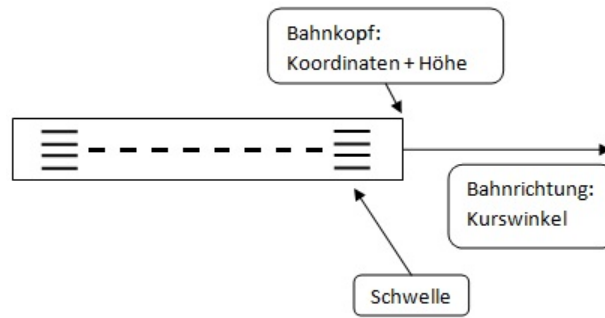


Abbildung 4.1: Start- und Landebahn

## 4.1.2 Objektstruktur

Für die Speicherung der Daten der Start- und Landebahnen werden neue Objekte erzeugt, da die in RouGe vorhandene Objektstruktur die neuen Informationen nicht aufnehmen kann.

**Runway** Das erste wichtige neue Objekt fasst alle Daten einer Runway zusammen. Dies umfasst den Namen der Bahn, die Länge und Breite, sowie die Entfernung der Schwellen vom Bahnkopf in Richtung Bahnmitte. Außerdem beinhaltet es auch noch die beiden Bahnköpfe und die Information, ob es sich um eine Lande- oder/und Startbahn handelt. Die Methoden in diesem Objekt bearbeiten diese Daten. Für die Rückgabe der Schwellen wird ein neues Bahnkopf-Objekt generiert, das die entsprechende Entfernung zum tatsächlichen Bahnkopf hat, aber sonst alle dessen Eigenschaften übernimmt.

**Bahnkopf und -schwelle** Das nächste Objekt definiert die Bahnköpfe oder Bahnschwellen. Es erbt vom RouGe-Objekt für dreidimensionale Wegpunkte. Das bedeutet, dass sich die Bahnköpfe wie Wegpunkte verhalten. Zusätzlich zu den Wegpunktinformationen benötigt dieses Objekt die Informationen über den Ausrichtungswinkel (Heading) der Bahn und bietet die Möglichkeit, die Bahnschwelle mit einem Endanflug-Profil zu verknüpfen. Bei der Zuweisung der Endanflug-Profile werden diese den Bahnschwellen zugewiesen. Eine Abfrage an das Objekt, ob es ein Profil zugewiesen bekommen hat, ist ebenfalls möglich.

**Airport Structure** Außerdem gibt es ein neues Objekt für die Flughafenstruktur. Diese ist eine Unterklasse der abstrakten Klasse *AbstractAirportStructure*. Diese abstrakte Klasse ist die Oberklasse für alle Flughafenstrukturen. Die verschiedenen Typen werden für verschiedene Module mit unterschiedlichen Anforderungen benötigt. Die gemeinsamen Anforderungen werden in der Oberklasse abgearbeitet. Außerdem gibt sie vor, welche Methoden die Unterklassen mindestens beinhalten müssen. Dies sind Methoden für die Deklaration und Abfrage des Standardbahntyps. Dieser definiert die Start- und Landebahnen nicht detailliert genug, sodass auch hier neue Objekte benötigt werden. Neu hinzugekommen ist eine HashMap, welche die Bahnen des Flughafens in Verknüpfung mit ihrem Namen abspeichert. Eine HashMap ist ein Assoziativspeicher nach dem Hashing-Verfahren. Es ist eine Tabelle, in der man einfach unsortiert Informationen speichern und mit einem assoziierten Schlüssel wiederverwenden kann. Für diese HashMap gibt es Bearbeitungsmethoden, welche die komplette HashMap bearbeiten oder zurückgeben, oder einzelne Einträge daraus. Neu sind die Methoden *getArrivalPath* und *getDeparturePath*. Die zweite davon gibt die Koordinaten der Startbahn als Abflugweg zurück, da in RouGe noch ein Abflugroutenkonzept fehlt. Mit der ersten der beiden Methoden bekommt man die Anflugroute zurückgegeben. Dazu wird der letzte Wegpunkt einer Flugroute benötigt

und der Flug selbst. Vom Flug wird die Landeschwelle benötigt, auf der das Flugzeug landet. In einer Liste wird der Landeanflug abgespeichert. Wenn die Landeschwelle ein Endanflug-Profil hat, wird daraus die Anflugroute vom letzten Wegpunkt aus abgerufen. Wenn kein Profil vorhanden ist, wird nur die Landeschwelle in die Liste eingetragen.

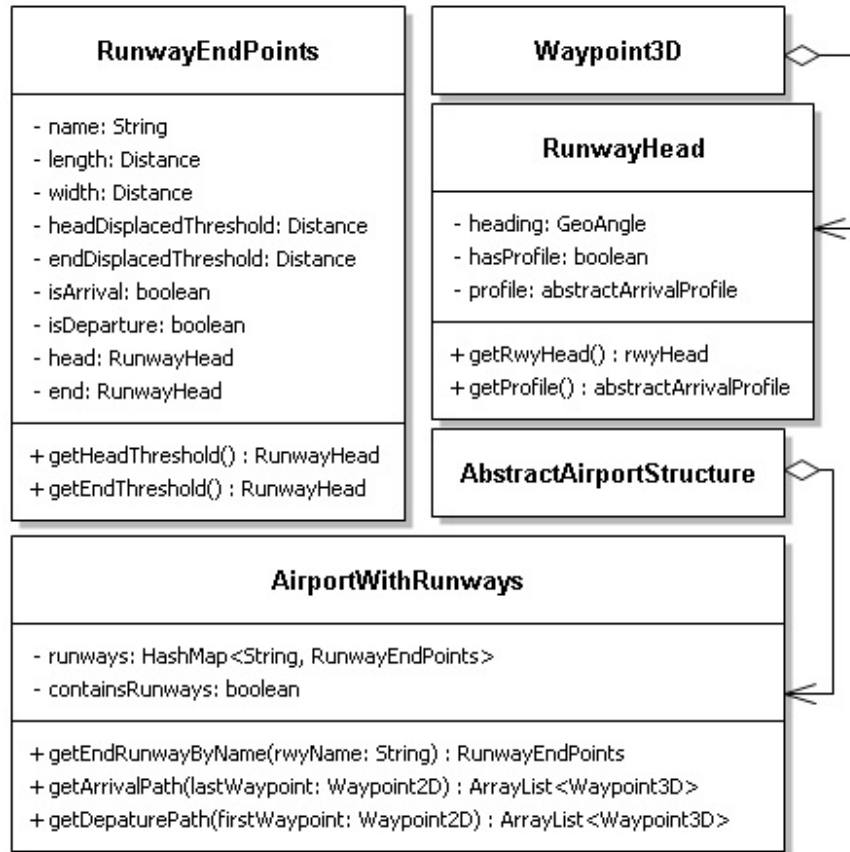


Abbildung 4.2: Neue Objekte mit wichtigen Methoden

### 4.1.3 Bahnauswahl

Im folgenden Abschnitt wird erklärt, wie die Bahnen für die einzelnen Flüge ausgewählt werden und wie diese Prozedur umgesetzt ist.

**Bahnen in Szenario einfügen** Die Methode *setRunways* hat die Aufgabe, eine HashTable mit Airport-Strukturen, die Runways enthalten, zu füllen. Eine Hashtable ist einer HashMap sehr ähnlich. Ein wesentlicher Unterschied ist bei der HashMap die Möglichkeit *null* als Schlüssel oder Wert zu nutzen. In dieser Hashtable wird der Name des Flughafens als Schlüssel verwendet und die Airport-Strukturen als Werte.

Zuerst werden die Daten der Nutzauswahl aus der Tabelle mit den Bahnen in die HashMap *dataMap* geschrieben. Danach wird in einer *for*-Schleife über alle Flüge eines Szenarios iteriert. In der Schleife wird jedem Flug eine Landebahn zugewiesen. Zuerst muss dazu überprüft werden, ob der Start-, beziehungsweise Zielflughafen als Schlüssel in der *dataMap* vorkommt. Für beide Typen funktioniert der Programmablauf gleich. Wenn der Flughafen vorhanden ist, wird mit der Methode *getRwyName* dem Flug eine Bahn zugewiesen. Dabei wird auch das Hashtable mit den Flughäfen ausgefüllt. Jede Bahn wird darin der richtigen Airport-Struktur zugewiesen.

Falls die Hashtable für einen Flughafen noch keinen Eintrag hat, wird ein neuer Eintrag angelegt.

**Bahn-Objekte aus Tabelleneinträgen erzeugen** In der Methode *tableToMap* wird die vom Nutzer bearbeitete Tabelle ausgelesen und die Daten in einer HashMap gespeichert. Dabei entspricht jede Reihe in der Tabelle einer Bahn. Für jede Bahn wird ein neues Objekt erzeugt. Die Informationen werden in diesen Objekten gespeichert. Die HashMap speichert die Bahnen flughafenweise ab. Das bedeutet, als Schlüssel wird der ICAO-Code des Flughafens verwendet und als Wert eine weitere HashMap, welche die Bahnen aufnimmt. Diese hat als Schlüssel den Bahnnamen, bestehend aus Flughafencode und den beiden Bahnkopfnamen.

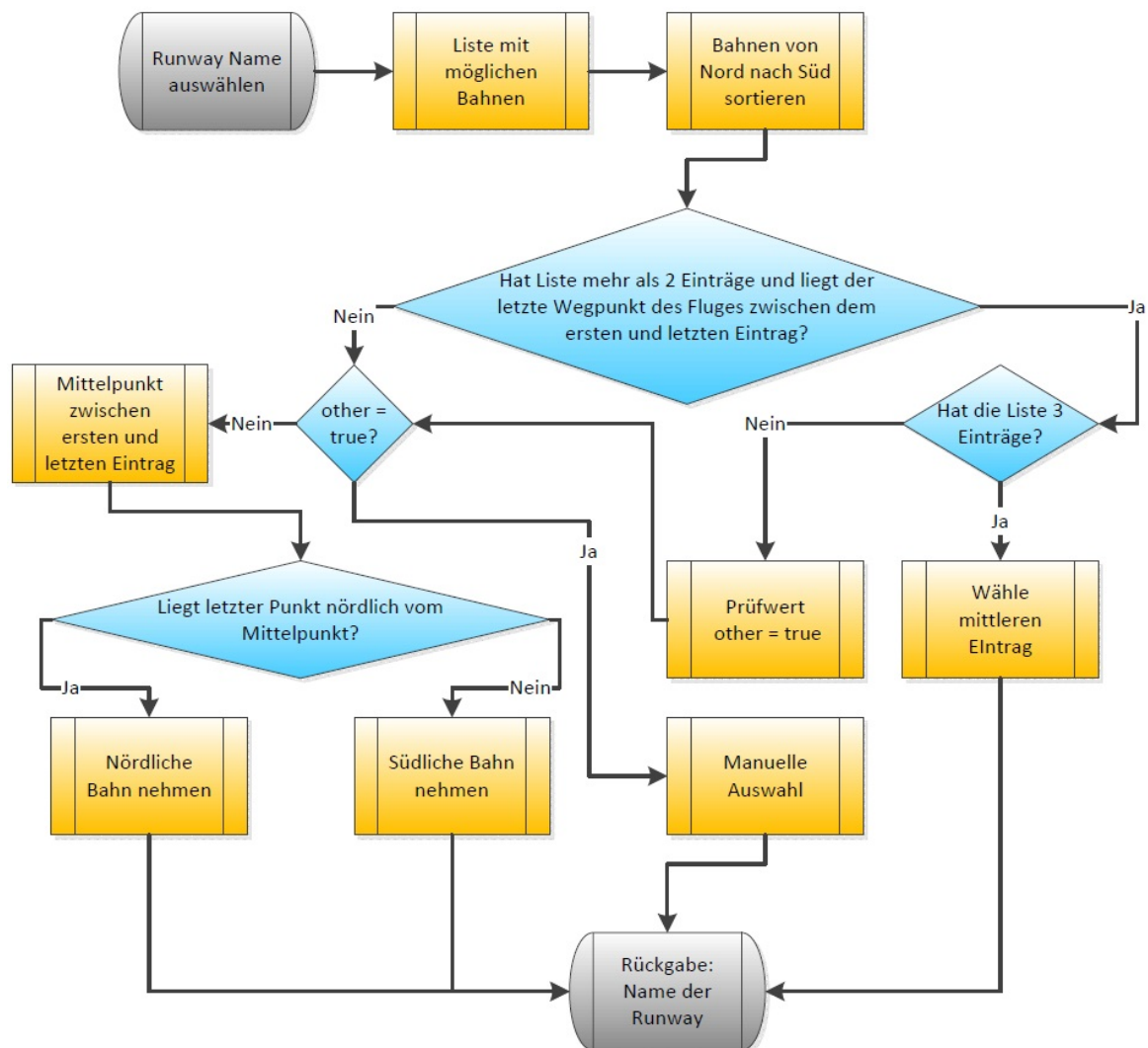


Abbildung 4.3: Auswahl der Namen für die Bahnen

**Bahnnamen festlegen - Automatische Bahnauswahl** Für die Zuweisung der Bahnen zu einem Flug wird in der Methode *getRwyName* eine automatische Auswahl gestartet. Sie erstellt zuerst zwei Listen mit allen Möglichkeiten für einen Flug. Eine Liste ist mit Startbahnen gefüllt und die andere mit Landebahnen. Beide Listen werden nach den Koordinaten der Bahnen von Nord nach Süd sortiert. Wenn eine Landebahn gesucht wird, so wird die dazugehörige Liste auf

ihre Anzahl an Einträgen geprüft. Wenn nur ein Eintrag vorhanden ist, wird dieser direkt als Landebahnname zurückgegeben.

Anderenfalls wird die Methode *chooseRwy* gestartet, welche die Bahnen anhand des Anflugbeziehungsweise Abflugweges aussucht. Hierbei werden die Positionen der ersten und der letzten Bahn in der Liste sowie der dichteste Wegpunkt der Flugroute verwendet. Es wird eine Prüfung vorgenommen, ob es mehr als zwei Möglichkeiten gibt und der letzte Wegpunkt sich zwischen den beiden Bahnen befindet. Wenn dieser Fall zutrifft und es nicht mehr als drei Bahnen zur Auswahl gibt, wird der mittlere Eintrag der sortierten Liste verwendet. Falls es mehr Bahnen gibt, wird ein Prüfwert auf wahr geschaltet, damit eine manuelle Auswahl stattfindet. Wenn keine Auswahl getroffen wurde und keine manuelle Auswahl notwendig ist, wird zwischen den beiden verbliebenen Bahnen gewählt. Dazu wird der geographische Breitengrad, der genau zwischen den beiden Bahnen liegt, genommen und überprüft, ob der Wegpunkt darüber oder darunter liegt. Für die jeweilige Auswahl wird der Name der Bahn zurückgegeben.

## 4.2 Benutzerinterface

Im folgenden Abschnitt geht es um die Schnittstellen zwischen dem neuen RouGe-Modul und dem Anwender. Es wird genauer darauf eingegangen, wie der Anwender die Daten ergänzt und parametrisiert und wie ihn das Programm dabei unterstützt.

Damit RouGe aus Flugplandaten Routen für die Simulationen erstellen kann, muss der Anwender in einem Filtermenü die Bedingungen für die Routengenerierung vorgeben. In diesem Menü kann man festlegen, ob bestimmte Wegpunkte ignoriert werden sollen und welche Flughäfen berücksichtigt werden. Außerdem bietet es einem die Möglichkeit, Voreinstellungen für spätere Exporte von Simulationsdaten für spezielle Simulatoren zu treffen. Bei der Aktivierung der Checkbox “Choose Runways from selected Airports” wird das Endanflug-Modul nach den Filtereinstellungen und vor der Routenberechnung gestartet. Beim Start des Moduls wird aus den Daten des Flughafenfilters eine Liste mit ausgewählten Flughäfen erstellt.

### 4.2.1 Tabellendarstellung Bahnen

Die Tabellendarstellung beginnt mit dem Einlesen der Daten für die Tabelle. Dazu sucht der Anwender die benötigte Quelldatei mit einem *JFileChooser* aus. Ein *JFileChooser* ist Dateiauswahldialog, der in der Swing-Grafikbibliothek von Java enthalten ist. “JFileChooser provides a simple mechanism for the user to choose a file.” [18] Die Datei wird mit einem einfachen Tabellenparser eingelesen. Dieser besteht aus zwei *for*-Schleifen. In der ersten werden die Spalten der Tabelle durchgegangen. Dabei wird die Headerzelle jeder Spalte überprüft. Wenn eine davon leer ist, wird diese Spalte ignoriert. Andernfalls wird die für jede Spalte die zweite Schleife gestartet, in der die Reihen durchgegangen werden. Hierbei wird zuerst für jede Reihe aus der Spalte mit den Flughafencodes der ICAO-Code überprüft, ob er in der Liste mit den ausgewählten Flughäfen vorkommt. Wenn ja, werden die Daten aus der Reihe in ein Datenarray gespeichert.

Nachdem alle benötigten Daten aus dem Quelldokument in das Datenarray geschrieben werden, wird daraus eine *JTable* erstellt. Die *JTable* kommt ebenfalls aus der Swing-Grafikbibliothek und erstellt Tabellen als Grafikobjekte. Dabei kann sie Arrays als Datenquelle verwenden. Die Headerdaten für die Tabelle werden in einem eigenen Array an die *JTable* übergeben, nachdem sie aus dem Datenarray herausgetrennt wurden.

Auf der Abbildung 4.4 sieht man die fertige Tabelle mit der Möglichkeit, in einer Spalte ein Drop-Down Menü zu verwenden. Ein Zelleneditor ermöglicht es, einzelne Zellen zu editieren



RwyFilter							
Arr or Dep?	airport_ident	length_ft	width_ft	le_ident	le_latitude_deg	le_longitude_deg	le_ele
Departure	EDDF	13123	197	07C	50.033	8.535	329
Arrival	EDDF	9186	148	07L	50.037	8.497	305
Arrival	EDDF	13123	148	07R	50.028	8.534	328
Departure	EDDF	13123	148	18	50.034	8.526	326
Arrival							
Departure							
Arrival and Departure							

Abbildung 4.4: Tabelle mit Bahninformationen

und fügt in diesem Fall für jede Zelle einer bestimmten Spalte das Swing-Grafikelement *JComboBox* hinzu. In diesem Menü stehen die verschiedenen Bahntypen zur Auswahl. Der Nutzer kann bestimmen, ob eine Bahn eine Start-, Landebahn oder beides ist. Der Zelleditor wird in einer eigenen Klasse definiert und den Zellen einzeln zugewiesen.

Das *JTable*-Objekt ist so eingestellt, dass einzelne Zellen bearbeitet werden können und Reihen beim darauf klicken markiert werden. Mit dem Befehl *table.getSelectedRows()* werden alle markierten Reihen abgefragt.

## 4.2.2 Parametrisierung der Endanflug-Profile

Nachdem die Daten über die Landebahnen verfügbar sind, öffnet sich ein neuer Frame, in dem der Nutzer den Landebahnen Endanflug-Profile zuweisen und parametrisieren kann. Dazu wählt der Nutzer in einer *JComboBox* zuerst die Landebahn aus und dann für diese Landebahn in einer zweiten *JComboBox* das Profil. Dann wird eine neue *JTable* erzeugt, die je nach Profiltyp vorgefertigte Einträge hat. Die Grundstruktur der Spalten ist für alle Profile gleich, sodass sich der Anwender nicht mehrere Eingabemethoden merken muss. Mit dem Knopf *Add Row* wird eine weitere Reihe in die Tabelle eingefügt, in der der Nutzer die Wegpunktliste erweitern kann. Mit dem *Apply*-Knopf werden die Einstellungen übernommen und auch auf dem lokalen Speicher gespeichert, damit die Daten wiederverwendet werden könne. Beim nächsten Laden des Dialogs werden die alten Daten automatisch erneut eingelesen.

Die *JTable* für die Parametrisierung werden auf die gleiche Art aufgebaut, wie die Tabelle für die Runways. Dieses *JTable* steht ein allgemeines Headerarray zur Verfügung, das für alle Profile verwendet wird und für jedes Profil ein Standard-Datenarray, das vor der Darstellung mit den lokal gespeicherten Daten abgeglichen wird.

RwyFilter
Approach Dialog

Now you can edit your Approach Settings.

EDDF\_07L\_25R
Point\_Merge

Name of Point	Distance	Unit	Direct Approach	Latitude	Longitude	Elevation
Final Approach Fix	14.7	nm	Trombone		null	5000.0
Merge Point	5.0	nm	Point_Merge		null	6000.0
Initial Approach Fix One	20.0	nm	308	null	null	12000.0
Initial Approach Fix Two	22.0	nm	8	null	null	10000.0

Add Row
Apply
☐

Abbildung 4.5: Tabelle mit Bahninformationen

## 4.3 Endanflug-Profile

In diesem Kapitel geht es um die Implementierung der Endanflug-Profile. Es wird näher darauf eingegangen, wie diese funktionieren und von RouGe verwendet werden.

### 4.3.1 Abstrakte Basisklasse

Damit RouGe alle Endanflug-Profile wie eine einzige Klasse behandeln kann, gibt es eine abstrakte Klasse. Die Profilklassen erben von dieser abstrakten Klasse und können alle als *AbstractArrivalProfile* angesprochen werden. Dazu müssen sie die von der abstrakten Klasse vorgegebenen Methoden implementieren. In diesem Fall ist dies die Methode *getArrivalPath*. So kann RouGe die Anflugroute bestehend aus einer Liste mit Wegpunkten mit dem Befehl *rwg.getProfile().getArrivalPath(lastWaypoint)* aus dem Endanflug-Profil einer Landebahn abrufen.

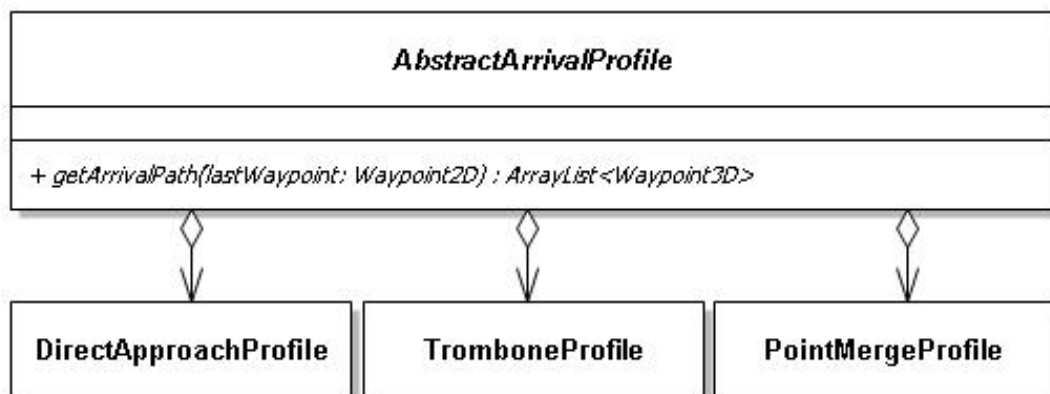


Abbildung 4.6: Abstrakte Profil Klasse

In Abbildung 4.6 sieht man die Struktur, wie die abstrakte Klasse eine Fassade für die Profilklassen bildet. RouGe spricht ausschließlich die abstrakte Klasse an, welche die Schnittstelle zwischen den Profilen und der Routengenerierung bildet. Das Endanflug-Modul erzeugt die Profile in den Profilklassen und weist sie den Landebahnen zu.

### 4.3.2 Profile

Nachdem der Anwender im Endanflug-Dialog auf *Apply* geklickt hat, werden die Profile erzeugt und in die Flughafenstrukturen zu den Landebahnen hinzugefügt. Eine Switch-Case Abfrage überprüft, welches Profil einer Landebahn zugewiesen wurde und erzeugt daraufhin das richtige Profil. Dazu muss dem Profil nur die Tabelle mit den Parametern übergeben werden.

Eine Gemeinsamkeit der Profile ist die individuelle Namensgebung der Wegpunkte. Jede Landebahn hat eine einzigartige Bezeichnung, bestehend aus dem ICAO-Code und den Bezeichnungen der Bahnköpfe. Diese bildet den Anfang der Namen. Daran wird die Bezeichnung aus der Tabelle gehängt. Beim Point-Merge Profil wird zusätzlich noch die Bezeichnung der Sequencing Leg angehängt. Durch diese Kombination entstehen einmalig vorkommende Namen, welche die Simulatoren unterscheiden können.

#### Direct Approach und Trombone

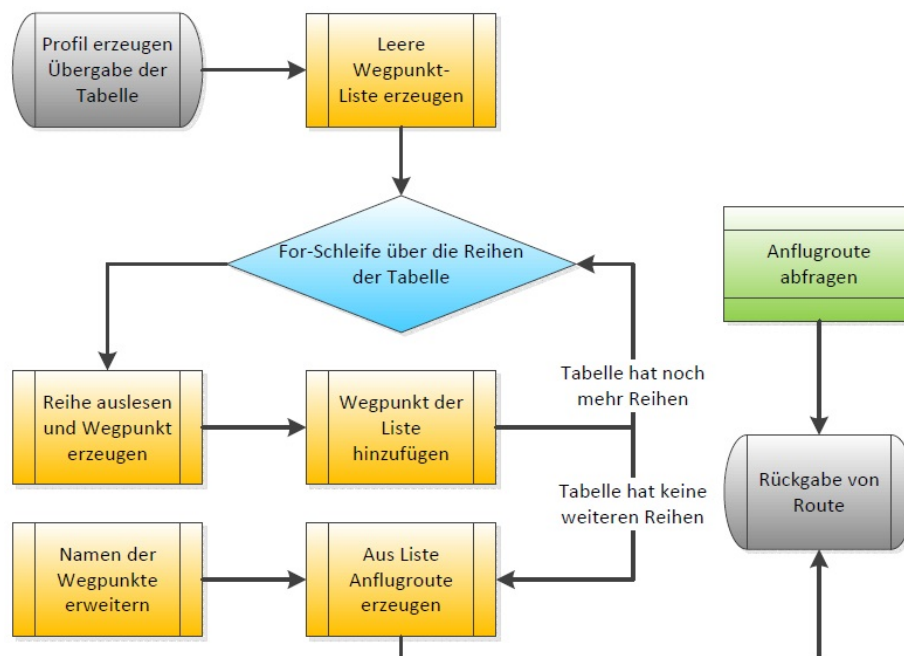


Abbildung 4.7: Ablauf Profilerzeugung bei Direct Approach und Trombone Profil

Bei den beiden Profilen für den Direct Approach und Trombone findet der gleiche Prozessablauf statt. Die Unterschiede liegen hier in der Beschreibung der Werte aus der Tabelle und den Vorgaben, die in der Tabelle enthalten sind. Für den Direct Approach wird mindestens ein Wegpunkt benötigt, während für den Trombone Anflug mindestens vier benötigt werden.

Für die Routenerzeugung wird zuerst eine leere Liste erzeugt, in der die Wegpunkte eingetragen werden. Dann werden alle Reihen der Tabelle durchgegangen, die dem Profil übergeben wurde. Da die Spaltentypen fest vorgegeben sind, können die Werte direkt ausgelesen und in Objekte zwischengespeichert werden. Eine Abfrage verhindert das Auslesen der Tabelle, wenn keine Koordinateninformationen vorhanden sind. Die Koordinaten werden aus der Entfernung zum vorherigen Punkt und dem Kurswinkel berechnet. Dabei ist der Anfangspunkt die Landeschwelle der Landebahn. In RouGe gibt es eine Bibliothek mit wichtigen Funktionen zur Kursberechnung. Die Entfernungsangaben können in unterschiedlichen Einheiten angegeben

werden. Dazu ist die passende Einheit in einer weiteren Spalte angegeben. In einer Switch-Case Abfrage wird die Einheit überprüft und daraufhin das richtige RouGe-Distance-Objekt erzeugt. Nachdem alle Werte in Objekte übertragen worden sind, wird ein *3DWaypoint* aus den berechneten Koordinaten, der Höhe und dem Namen erzeugt und der Liste angehängt. Diese Liste wird im Profil der Landebahn gespeichert und bei der Abfrage *getArrivalPath* übergeben.

Now you can edit your Approach Settings.

EDDF\_07L\_25R ▼ Trombone ▼

Name of Point	Distance	Unit	Angle	Latitude	Longitude	Elevation
Final Approach Fix	5.0	nm				5000.0
Point1	5.0	nm				10000.0
Point2	5.0	nm				10000.0
Initial Approach Fix	5.0	nm				10000.0
Point3	5.0	nm				12000.0

Abbildung 4.8: Standard Werte und Vorgaben - Trombone

**Trombone** Die Tabelle in Abbildung 4.8 zeigt die Tabellenstruktur der Parameter für die Wegpunktangabe der Profile. Sie beinhaltet den Namen des Wegpunktes, die Distanz zum vorherigen Wegpunkt, zusammen mit der Maßeinheit. Dazu kommen noch der Kurswinkel und die Höhenangabe am Ende der Tabelle. Alternativ lassen sich die Wegpunkte als Koordinaten angeben. Für das Trombone-Profil sind fünf Punkte in der Tabelle schon vorgegeben und mit Standardwerten befüllt. Diese Werte orientieren sich an der Anflugkarte für den Trombone-Anflug auf Frankfurt am Main aus dem Luftfahrthandbuch der Deutsche Flugsicherung (DFS).[6] Die Standardwerte umfassen den Final Approach Fix, den beiden Eindrehpunkten (Punkte B und C auf Abbildung 3.4), dem Initial Approach Fix und einem weiteren Wegpunkt, von dem aus die Flugzeuge auf die Anflugroute geleitet werden.

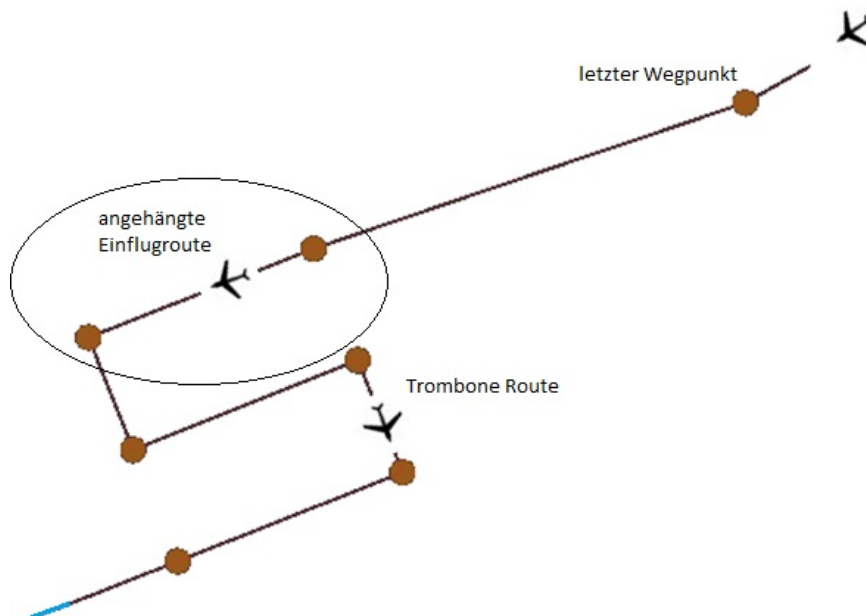


Abbildung 4.9: Trombone Anflug

In Abbildung 4.9 kann man die erzeugte Trombone-Anflugstruktur erkennen. Da der Flug nicht aus der Anflugrichtung der Trombone Struktur kommt (siehe letzter Wegpunkt der Route), muss dieser noch über eine Einflugroute fliegen. Diese wird durch zusätzlich angegebene Wegpunkte erzeugt und leitet das Flugzeug auf die Trombone-Anflugstrecke. Die erzeugte Trombone-Anflugroute besteht aus dem minimalen Weg, der für dieses Anflugverfahren machbar ist.

Now you can edit your Approach Settings.

EDDF\_07L\_25R ▼ Direct\_Approach ▼

Name of Point	Distance	Unit	Angle	Latitude	Longitude	Elevation
Final Approach Fix	5.0	nm				5000.0

Abbildung 4.10: Standard Werte und Vorgaben - Direct Approach

**Direct Approach** In der Abbildung 4.10 ist ein Standardwert für den Final Approach Fix vorgegeben. Der Final Approach Fix findet sich normalerweise in drei bis sieben nautischen Meilen Entfernung zur Landeschwelle. [4, S.155] Fünf nautische Meilen ist der Mittelwert des Standardbereiches.

In Abbildung 4.11 sieht man den Direct Approach. Dieser besteht in diesem Fall nur aus dem *Final Approach Fix*. Der Anwender kann bei dieser Struktur auch weitere Wegpunkte angeben, welche die Anflugroute verlängern würden.

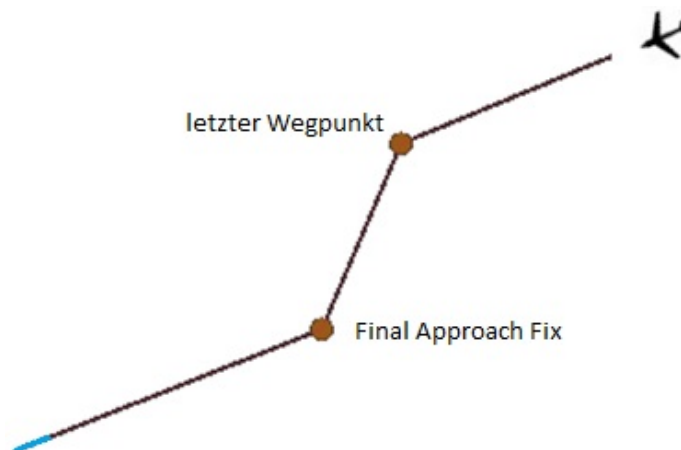


Abbildung 4.11: Direct Approach

## Point-Merge

Das Point-Merge-Profil verlangt eine komplizierte Herangehensweise, da verschiedene Anflugrouten erzeugt werden müssen. Das Auslesen der Tabelle funktioniert auf die gleiche Weise wie bei den anderen beiden Profilen, jedoch werden die Wegpunkte nicht direkt in eine Liste getragen, sondern als globale Variablen zwischengespeichert, da der Anflugweg individuell berechnet werden muss. Dies bedeutet auch, dass keine zusätzlichen Wegpunkte an die Struktur angehängt werden können.



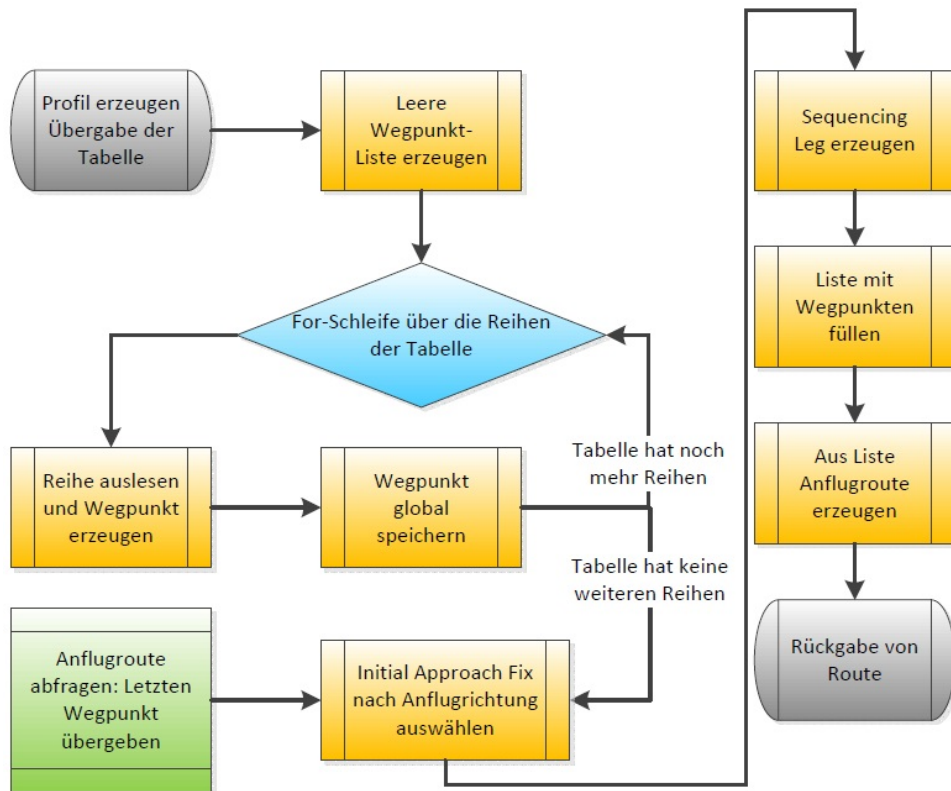


Abbildung 4.12: Ablauf Profilerzeugung beim Point Merge Profil

Bei der Abfrage der Anflugroute wird der letzte Wegpunkt mit übergeben. Dieser gibt die Richtung vor, aus der ein Flug kommt. Anhand der Anflugrichtung wählt das Point-Merge Profil dann aus, welchen Einflugpunkt das Flugzeug nimmt. Als nächstes wird die Sequencing Leg berechnet. Dazu wird der Endpunkt dieser Strecke benötigt, an der das Flugzeug spätestens zum Merge-Punkt eindreht. Dieser wird aus den Informationen der beiden Einflugpunkte berechnet. Dazu wird die Entfernung zum Merge-Punkt und die Höhe vom tatsächlichen Einflugpunkt verwendet und der Kurswinkel des nicht verwendeten Einflugpunktes. Nun werden die beiden Punkte der Sequencing Leg, des Merge-Punktes und des Final Approach Fix nacheinander in eine Liste eingetragen, um eine Anflugroute zu bilden. Die Liste ist aber noch nicht vollständig. Die Sequencing Leg muss noch um Zwischenpunkte erweitert werden, damit ein Kreisausschnitt entsteht. Dazu wird der Zwischenwinkel zwischen dem Einflug- und dem Endpunkt genommen und in zehn Teile geteilt. Dieses Zehntel vom Teilstück wird auf den Kurswinkel vom Einflugpunkt gerechnet und zwischen den Endpunkten in die Liste eingetragen. Für den nächsten Zwischenpunkt wird der Winkel vom vorherigen um ein weiteres Zehntel erweitert und in vor den Endpunkt in die Liste getragen. Dieser Schritt wird neun mal wiederholt, sodass die Sequencing Leg nun aus elf Wegpunkten besteht und ein Kreisausschnitt sichtbar wird.

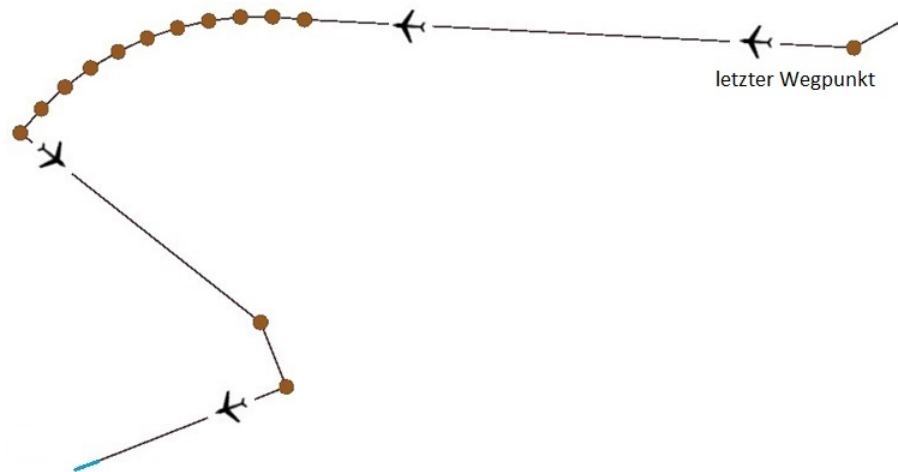


Abbildung 4.13: Point-Merge Anflug

Abbildung 4.13 zeigt einen Point-Merge Anflug, bei dem die Flugzeuge vom letzten Wegpunkt über die Sequencing Leg zum Merge-Punkt und dann zur Landebahn fliegen.

Now you can edit your Approach Settings.

EDDF\_07L\_25R ▼ Point\_Merge ▼

Name of Point	Distance	Unit	Angle	Latitude	Longitude	Elevation
Final Approach Fix	10.0	nm				5000.0
Merge Point	5.0	nm				6000.0
Initial Approach Fix One	20.0	nm				12000.0
Initial Approach Fix Two	22.0	nm				10000.0

Abbildung 4.14: Standard Werte und Vorgaben - Point-Merge

Die Tabelle 4.14 zeigt die Standardwerte für den Point-Merge Anflug. Die für die Berechnung notwendigen Punkte sind neben dem Final Approach Fix der Merge-Punkt und die beiden Einflugpunkte. Die Werte sind Testwerte, welche aus dem mittleren Bereich real existierender Point-Merge Strukturen stammen. [14]

# Kapitel 5

## Beschreibung der Ergebnisse

Mit dieser Bachelorarbeit ist in RouGe ein Modul zur Generierung von Endanflug-Profilen entstanden. Mit diesem Modul erweitern die Anwender die Flugpläne um Informationen zu Start- und Landebahnen und erzeugen flexible Endanflug-Profile. Die Programmtests wurden nach den einzelnen Implementierungsschritten mit Flugplandaten von EUROCONTROL durchgeführt. Im folgenden Kapitel werden die Testdaten vorgestellt und beschrieben, was RouGe mit dem neuen Modul erzeugt.

### 5.1 Vorstellung der Testdaten

Die Testdaten bestehen aus Flugplänen, die Informationen zu den Flugrouten, Flugzeiten und ICAO-Codes der Flugzeugtypen und Flughäfen beinhalten. Die dargestellten Flugrouten bestehen aus mehreren Routenpunkten vom Start bis zum Ziel einer Route mit den geografischen Koordinaten, der Punkte. RouGe wurde in der Testphase mit verschiedenen realen Flugplandaten geladen. Diese hatten einen Umfang von einer einzelnen Flugroute mit nur wenigen Routenpunkten bis hin zu mehreren hundert hoch aufgelösten Flugrouten. Die Flugpläne liegen in Form von DDR-Daten vor. Das Demand Data Repository (DDR) ist ein Projekt von Eurocontrol mit dem Ziel, Flugplandaten bereitzustellen, die für die Nutzung im Rahmen von zukünftiger Luftraumgestaltung und der Analyse von vergangenen Trends geeignet sind. “The objective of DDR is to provide European airspace planners and airspace users with an accurate picture of past and future European air traffic demand, to meet their planning and monitoring needs.”[16]

Die für die Tests verwendeten Parameter sind Testwerte und kommen so nicht in der Realität vor.

### 5.2 Was macht das Modul für Endanflug-Profile?

Das Modul für Endanflug-Profile ist ein neues Routengenerierungsmodul für den Routengenerator (RouGe). Mit den Daten über die Start- und Landebahnen kann der Nutzer die Szenarien der Flugplandaten um Flughäfen erweitern und an die Flugrouten verschiedene Anflugrouten anhängen.

Der Nutzer startet das Modul bei den Filtereinstellungen für die Berechnung der Routen durch die Aktivierung der Checkbox “Choose Runways from selected Airports”. Nach Beendigung der Filtereinstellungen öffnet sich ein neues Fenster, in dem zuerst die Tabelle mit den Runway-Informationen eingelesen werden muss. Daraufhin werden die Runways in einer Tabelle dargestellt, dessen Flughäfen der Nutzer im Filtermenü ausgewählt hat. In der Tabelle



kann der Nutzer noch festlegen, ob die Bahn als Start- oder Landebahn oder beides genutzt wird. Wenn ein Flughafen nur eine Bahn hat, wird automatisch der Typ auf Start- und Landebahn gesetzt. Der Nutzer kann nun einzelne oder mehrere Bahnen für die Weiterverwendung markieren. Beim Betätigen des *Apply*-Knopfes werden die Daten übernommen und ein weiteres Fenster öffnet sich. Der Nutzer wird nun aufgefordert, die Endanflug-Profile für die ausgewählten Landebahnen einzugeben. Dazu gibt es zwei Menüs. Im ersten Menü wird die zu bearbeitende Landebahn ausgesucht und im Zweiten ein Endanflug-Profil. Nach der Auswahl in den beiden Menüs öffnet sich eine Tabelle, in jene die nötigen Parameter eingetragen werden müssen. In die Tabelle werden die fixen Wegpunkte der Endanflug-Profile eingetragen. Benötigt werden entweder die Koordinaten, oder die Entfernung und der Kurswinkel zum vorherigen Wegpunkt. Bei letzterem ist die Landeschwelle der erste Orientierungspunkt. Alle geänderten Daten werden zur Wiederverwendung auf dem lokalen Computer gespeichert und beim nächsten Einstellungsprozess wiederverwendet.

Mit den festgelegten Parametern erstellen die Profile flexibel für die einzelnen Flüge die Anflugrouten ab dem letzten Wegpunkt bis zur Landebahn. Dabei werden die Flüge auch automatisch auf mehrere Bahnen aufgeteilt, falls ein Flughafen mehrere besitzt. Die Zuweisung der Bahnen erfolgt nach Anflugrichtung.

Abbildung 5.1 und 5.2 zeigen die Visualisierungen des direkten Anfluges und des Trombone Anfluges auf den Flughafen Frankfurt am Main. Es wird jeweils ein Ausschnitt von 50 nautischen Meilen Radius um den Flughafen gezeigt. Die grünen Routen sind dabei die startenden Flüge, die noch keine besonderen Abflugwege haben. Die roten Routen sind die Anflugrouten der Flugzeuge auf die beiden Landebahnen des Flughafens. Das Direct-Approach Profil erzeugt eine Abfolge von Wegpunkten ab den Landebahnen, welche die Flugzeuge abfliegen. Zu sehen ist der Direct Approach auf der Abbildung 5.1. Die Abbildung 5.2 zeigt den Trombone-Anflug, bei dem die Flugzeuge die beiden Trombone-Strukturen abfliegen. Die Abbildungen 5.1 bis 5.3 zeigen Visualisierungen, die mit RouGe erstellt worden sind.

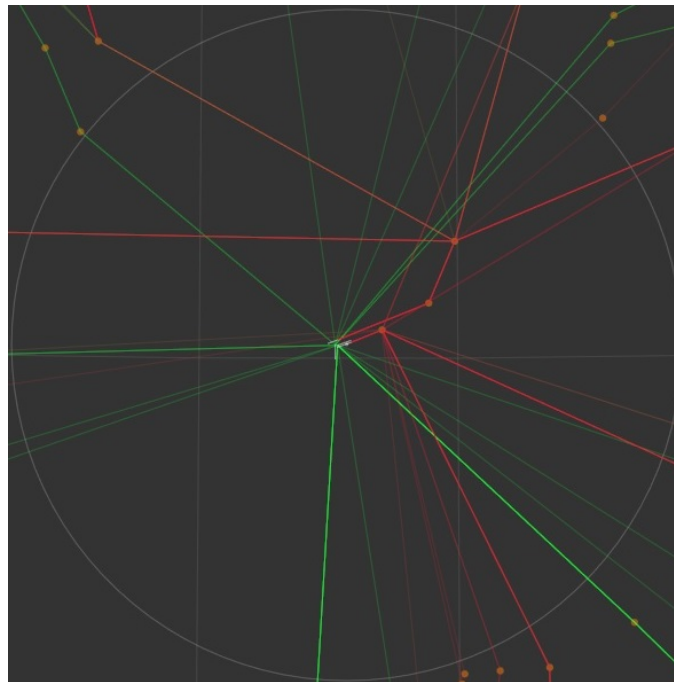


Abbildung 5.1: Direct Approach Anflug

Die Abbildung 5.3 zeigt das Point-Merge Anflugverfahren. Das Bild hat die gleichen Merk-

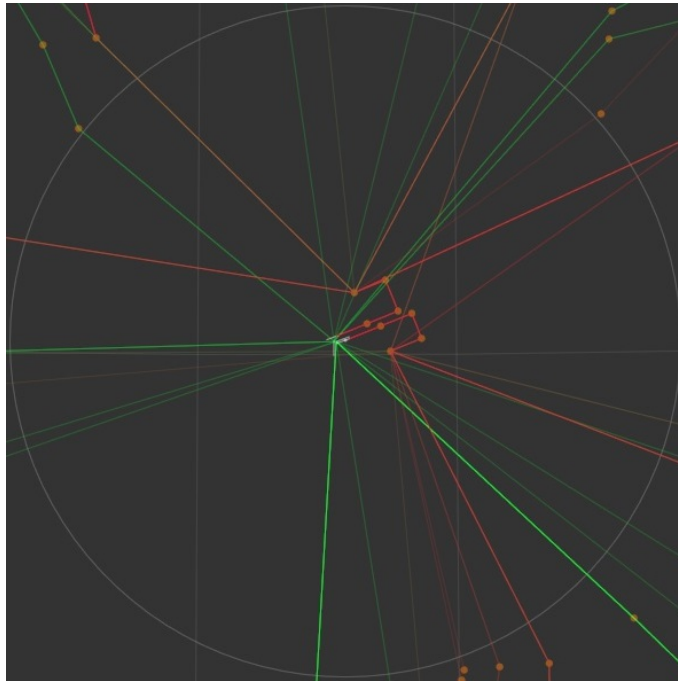


Abbildung 5.2: Trombone Anflug

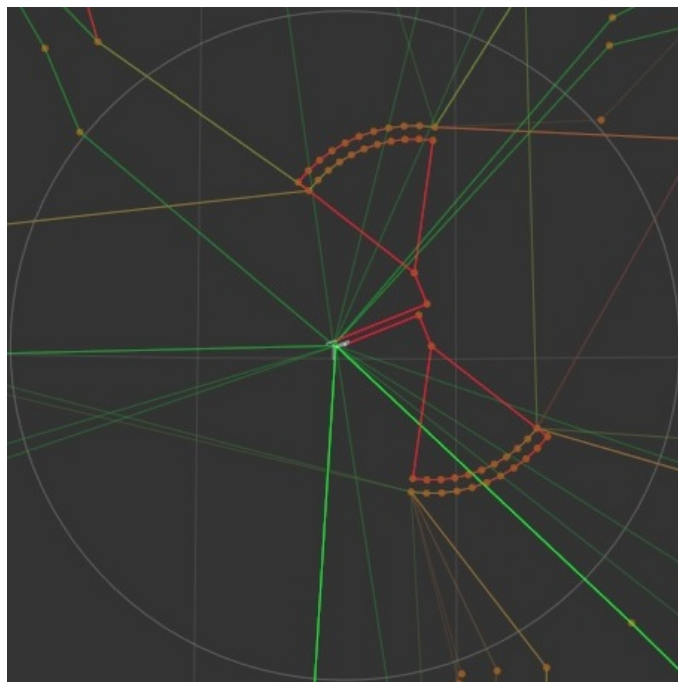


Abbildung 5.3: Point-Merge Anflug

male wie die Abbildungen 5.1 und 5.2. Das Point-Merge Profil erzeugt pro Landebahn zwei gegenläufige Sequencing Legs. Diese besitzen einen seitlichen und einen vertikalen Abstand zueinander. Für die Berechnung der Sequencing Legs werden die beiden Einflugpunkte und der Merge-Punkt benötigt. Daraus werden die Endpunkte der beiden Sequencing Legs und die Zwischenpunkte berechnet. Zusätzlich findet auch eine Entscheidung statt, welche Anflugroute ein Flug nimmt. Dies wird wie bei der Landebahnauswahl anhand der Anflugrichtung entschieden.

### 5.2.1 Ergebnisse in AEDT

Da die Anflugrouten als Wegpunkte an die Flugrouten angehängt werden, ist der Export in andere Programme ohne Änderungen an den Exportmodulen möglich. Auf den Abbildungen 5.4, 5.5 und 5.6 sieht man die Studien, die in die Auswertungssoftware AEDT eingelesen wurden. Auf den Ausschnitten erkennt man jeweils eine Route eines Fluges, der eine Anflugstruktur durchfliegt. Dabei gibt der schwarze Pfeil jeweils die Flugrichtung an. Auf den Abbildungen für den Direct Approach und den Trombone Approach erkennt man außerdem noch einen rosa Punkte Teppich. Dieser markiert jeweils ein Gebiet, in dem Emmisions- und Lärmstudien durchgeführt werden sollen. Die Flugrouten entsprechen den Anpassungen, die im neuen RouGe-Modul vorgenommen wurden. Abbildung 5.4 zeigt die ganze Oberfläche von AEDT.

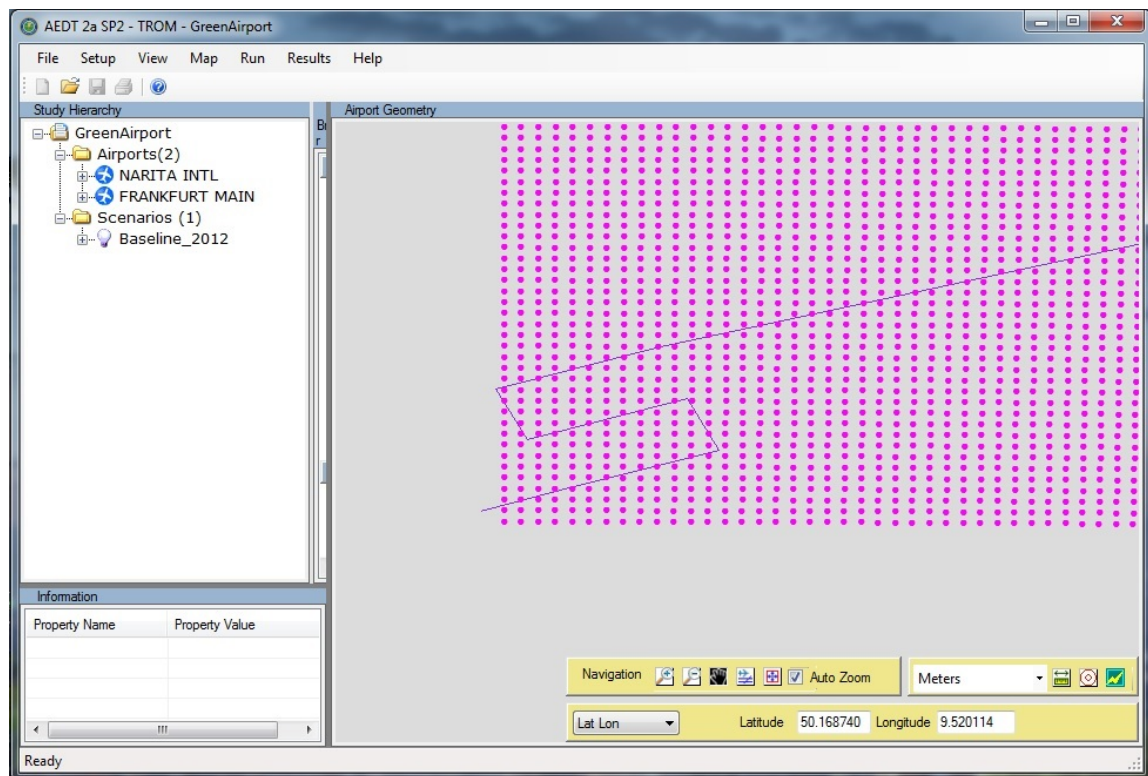


Abbildung 5.4: AEDT: Trombone Anflug

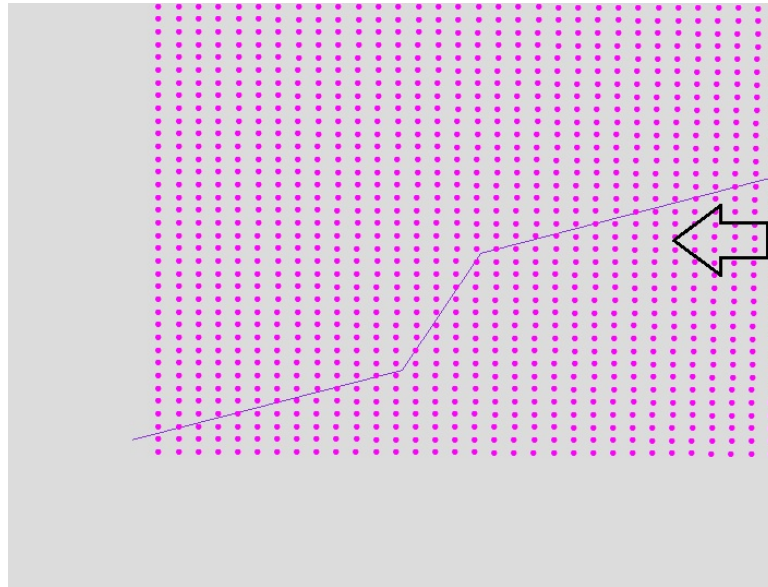


Abbildung 5.5: AEDT: Direct Approach

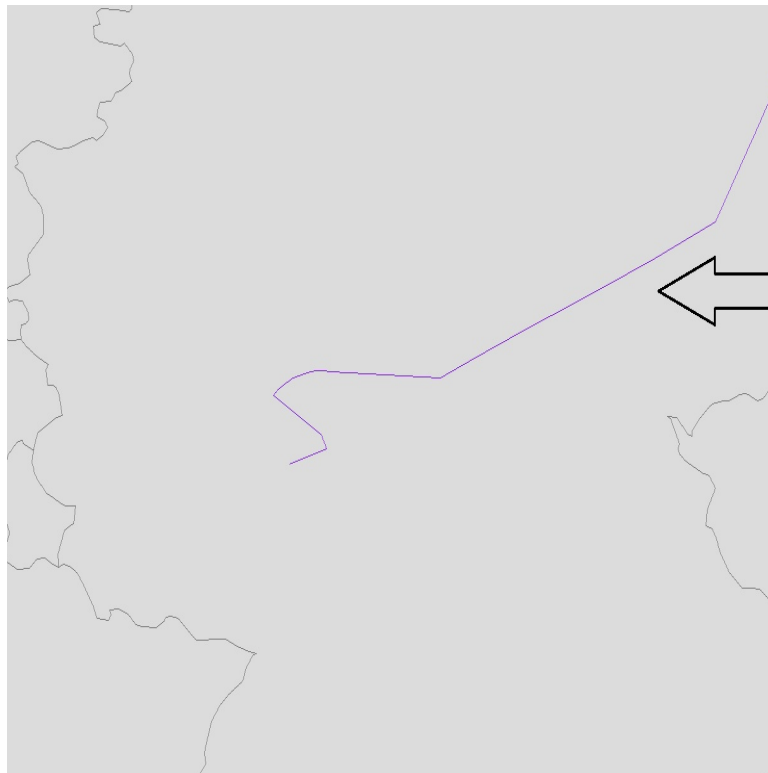


Abbildung 5.6: AEDT: Point-Merge Anflug

## 5.2.2 Ergebnisse in AirTOP

Auch für den Schnellzeitsimulator AirTOP gibt es ein RouGe-Exportmodul. Dies erzeugt die Routendaten und Flugpläne für ein Szenario in AirTOP. Die folgenden Abbildungen zeigen die Anflugstrukturen in diesem Schnellzeitsimulator. Man sieht auf ihnen viele grüne Routen und jeweils eine blaue. Die blaue Route ist eine ausgewählte Route, die in AirTOP genauer betrachtet wird. Die weißen Pfeile geben die Flugrichtung dieser Flüge an. Die grünen Routen sind weitere Routen, auf die kein Fokus gesetzt ist.

Abbildung 5.7 zeigt den Anflug über eine Trombone-Struktur auf den Flughafen Frankfurt am Main aus südlicher Richtung.

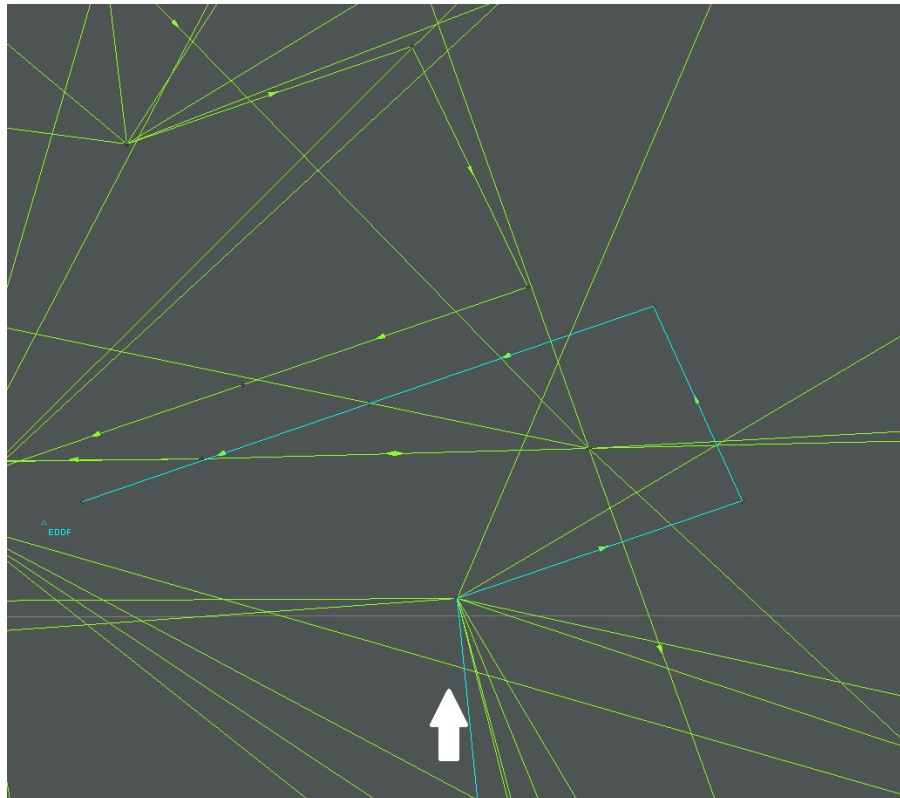


Abbildung 5.7: AirTOP: Frankfurt Süd Trombone-Anflug

Abbildung 5.8 und 5.9 zeigen beide den Point-Merge Anflug auf den Flughafen München. Der fokussierte Flug im ersten Bild kommt aus Richtung Nordwest und fliegt über die innere Sequencing Leg zum Flughafen. Der Flug im zweiten Bild kommt aus Richtung Nordost und fliegt über die äußere Sequencing Leg.



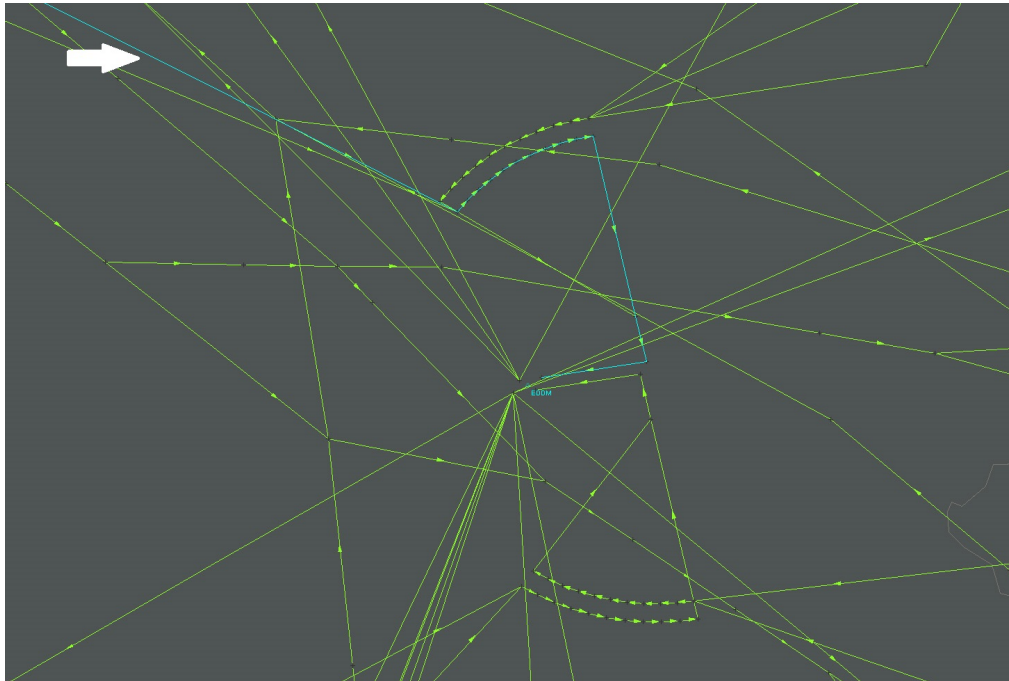


Abbildung 5.8: AirTOP: München Northwest Point-Merge Anflug

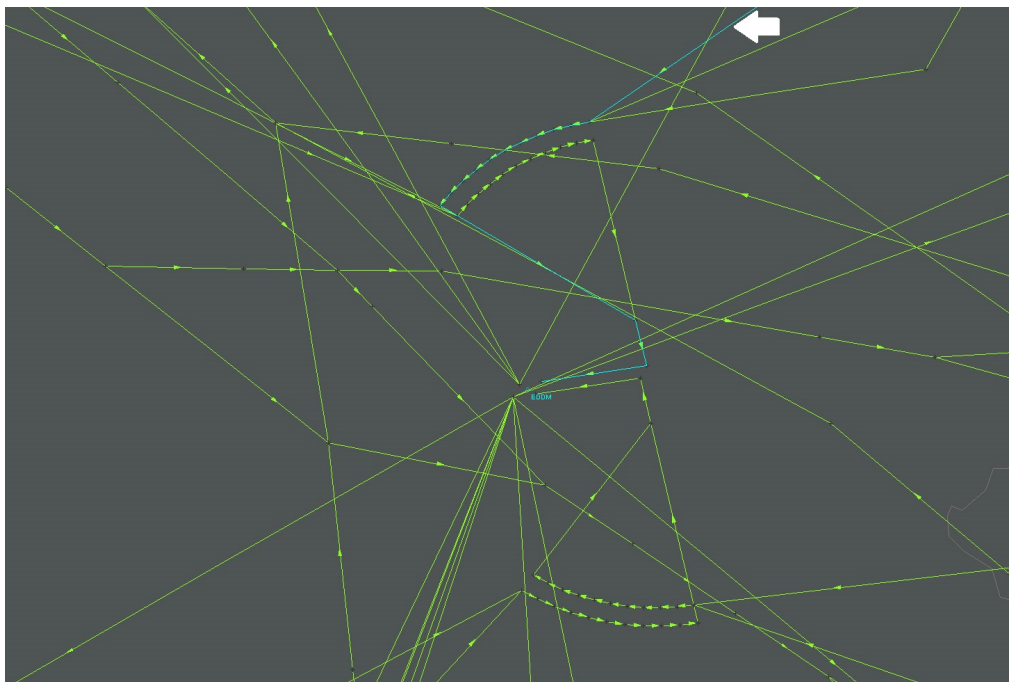


Abbildung 5.9: AirTOP: München Nordost Point-Merge Anflug

# Kapitel 6

## Diskussion

In diesem Abschnitt der Bachelorarbeit werden Ergebnisse der Arbeit diskutiert und beschrieben, welche Zielvorgaben erreicht wurden und welche Arbeitsschritte Probleme hervorbrachten. Die Zielvorgabe dieser Arbeit ist die Entwicklung und Implementierung eines Konzepts zur flexiblen Generierung von Endanflug-Profilen für den Routengenerator (RouGe). Zu allen Aufgabenpunkten der Aufgabenstellung wurde eine Lösung erarbeitet.

Das Modul für Endanflug-Profile ist ein Routengenerierungsmodul, das die Erzeugung von Trajektorien-Abschnitten für verschiedene Anflugrouten ermöglicht und an bestehende Flugrouten anhängt. Durch geeignete Parametrisierung können drei verschiedene Anflugverfahren betrachtet werden. Weitere Anflugverfahren können einfach implementiert werden, da alle Anflugverfahren von einer abstrakten Klasse erben und RouGe nur diese abstrakte Klasse sieht.

Der Programmablauf wurde um neue Funktionen ergänzt, kann weiterhin aber auch ohne das neue Modul durchlaufen werden. Die Datenstruktur musste um einige neue Objekte erweitert werden, da die bereits vorhandenen Objekte die benötigten Funktionen nicht bieten. Damit sich der Anwender leicht zurechtfindet, durchläuft er die Benutzeroberfläche des Moduls schrittweise und wird bei jedem Schritt um wenige Aktionen gebeten. Zusätzlich wird der Nutzer dahingehend unterstützt, dass eingegebene Parameter lokal gespeichert werden und wiederverwendet werden können, ohne sie erneut einzugeben.

Nach der Parametrisierung erzeugen die Profile neue Anflugstrecken für die Flüge, die ohne Anpassung des Visualisierungsmoduls dargestellt werden können. Lediglich die Darstellung der Runways musste implementiert werden.

Einschränkungen gibt es bei der Zuweisung der Landeswellen. Zurzeit findet noch keine Zuweisung der Flüge nach Windrichtung statt. Die Flüge landen immer auf der gleichen Landeschwelle einer Bahn. Außerdem war die Namensgebung der Wegpunkte eine Herausforderung, da die verschiedenen Exporte genaue Kriterien für die Wegpunktnamen haben.

# Kapitel 7

## Fazit und Ausblick

Mit dieser Bachelorarbeit ist ein wichtiges Modul für den Routengenerator (RouGe) entstanden, das die Flugplandaten um Runways und Anflugrouten erweitert. Es erleichtert die Modellierung von Szenarien für Simulationsdurchläufe, da es die Generierung von Anflugverfahren flexibel macht und automatisiert. Es können nun Anflugverfahren direkt im automatisierten Modellierungsprozess hinzugefügt werden, ohne sie nachträglich manuell einzufügen. Das Modellieren der Anflugrouten geschieht nun in wenigen Minuten, anstatt vieler Stunden oder Tage, je nach Umfang der Studien.

Die Aufgabenstellungen und darauf aufbauenden Anforderungen für dieses Projekt wurden alle erfüllt. Im Rahmen dieser Arbeit wurde ein Konzept erstellt und implementiert, das für den Routengenerator (RouGe) eine flexible Generierung von Endanflugprofilen ermöglicht. Dies umfasst den Ersatz bestehender Trajektorieninformationen wie auch die Ergänzung von Trajektorien um geeignete Endanflüge. Es können sowohl aktuelle wie auch mögliche zukünftige Endanflug-Profile erzeugt werden.

Durch Literaturrecherche wurden die Eigenschaften verschiedener Anflug-Konzepte herausgearbeitet und verglichen. Darauf aufbauend ist ein Konzept entstanden und implementiert worden, welches die Erzeugung von Trajektorien-Abschnitten für verschiedene Anflugverfahren ermöglicht. Durch geeignete Parametrisierung können drei verschiedene Verfahren betrachtet werden. Die Exporte, die mit dem neuen Modul und RouGe erstellt wurden, konnten in AEDT und AirTop eingelesen und verwendet werden. Des Weiteren kann RouGe nun auch Informationen zu Start- und Landebahnen einlesen, verarbeiten und visualisieren.

Im Laufe der Konzeptentwicklung und Implementierung sind viele Möglichkeiten aufgetaucht, RouGe und das neue Modul um weitere Funktionen zu erweitern. Zum Beispiel fehlt eine sinnvolle Abflugstruktur für die Flugrouten und die Selektion der Bahnschwellen pro Bahn existiert ebenfalls noch nicht. Dies kann durch die Implementierung von Windrichtungen erreicht werden. Eine weitere Aufgabe für die Zukunft ist es, die Anflugrouten für AirTop speziell als Approach Legs zu kennzeichnen, damit damit weitere Vereinfachungen im Umgang mit AirTop gewährleistet sind.

Das Ziel für die Bachelorarbeit war es, ein wissenschaftliches Modul für RouGe zu entwickeln und dieses Ziel wurde erreicht.



# Literaturverzeichnis

- [1] Sulkowski P.: Entwurf und Implementierung einer flexiblen Luftraumstruktur mit graphischer Routendarstellung für den Routengenerator RouGe, Braunschweig 2013
- [2] Koopmann, J./ Barberio, G./ Ahearn, M./ Hwang, S.: Aviation Environmental Design Tool (AEDT) 2a Service Pack 1 User Guide, Cambridge: U.S. DOT Volpe Center, 2012
- [3] Kern, S./ Knabe, F.: Modellbasierte Validierung zukünftiger ATM Konzepte/Verfahren mit Schnellzeitsimulation  
Braunschweig: Deutsches Zentrum für Luft- und Raumfahrt, 2014
- [4] Klußmann, N./ Malik, A.: Lexikon der Luftfahrt,  
Heidelberg: Springer-Verlag, 2012
- [5] Mensen, H.: Handbuch der Luftfahrt,  
Heidelberg: Springer-Verlag, 2003
- [6] Deutsche Flugsicherung: DFS Luftfahrthandbuch Deutschland  
Langen: DFS Deutsche Flugsicherung GmbH, 2014
- [7] Deutsche Flugsicherung: Pressemitteilung  
"DFS prüft innovatives Anflugverfahren 'Point Merge' "  
Langen: DFS Deutsche Flugsicherung GmbH, 22.Dezember 2011
- [9] Ullenboom, C. (2010): Java ist auch nur eine Insel  
Bonn: Galileo Press GmbH, 9. Auflage

# Quellenverzeichnis

- [10] Schultz, M.: Luftverkehrssysteme,  
<http://www.dlr.de/fl/desktopdefault.aspx/tabid-1136/>  
Stand: 28.08.2013
- [11] Quin, L.: Extensible Markup Language (XML), Introduction,  
<http://www.w3.org/XML/>  
Stand: 04.03.2014
- [12] o.A.: Was ist die Java-Technologie und wozu wird sie benötigt?,  
[http://www.java.com/de/download/faq/whatis\\_java.xml](http://www.java.com/de/download/faq/whatis_java.xml)  
Stand: 28.08.2013
- [13] Deutsche Flugsicherung: Simulation zur Validierung neuartiger Verfahren  
[http://www.dfs.de/dfs\\_homepage/de/Flugsicherung/Forschung%20&%20Entwicklung/Aufgaben%20&%20Ergebnisse/Simulationen%20zur%20Validierung%20neuartiger%20Verfahren/](http://www.dfs.de/dfs_homepage/de/Flugsicherung/Forschung%20&%20Entwicklung/Aufgaben%20&%20Ergebnisse/Simulationen%20zur%20Validierung%20neuartiger%20Verfahren/)  
Stand: 15.08.2014
- [14] EUROCONTROL: Point Merge: improving and harmonising arrival operations with existing technology  
<https://www.eurocontrol.int/services/point-merge-concept>  
Stand: 09.09.2014
- [15] ICAO: About ICAO,  
<http://www.icao.int/about-icao/Pages/default.aspx>,  
Stand: 21.03.2014
- [16] Eurocontrol: Demand Data Repository (DDR) project,  
<http://www.eurocontrol.int/ddr>,  
Stand: 24.03.2014
- [17] Megginson Technologies Ltd.: OurAirports  
<http://ourairports.com/data/#excel>  
Stand: 02.09.2014
- [18] Oracle: Class JFileChooser  
<http://docs.oracle.com/javase/7/docs/api/javax/swing/JFileChooser.html> Stand: 08.09.2014